80 microcomputing
the magazine for TRS-80™ users

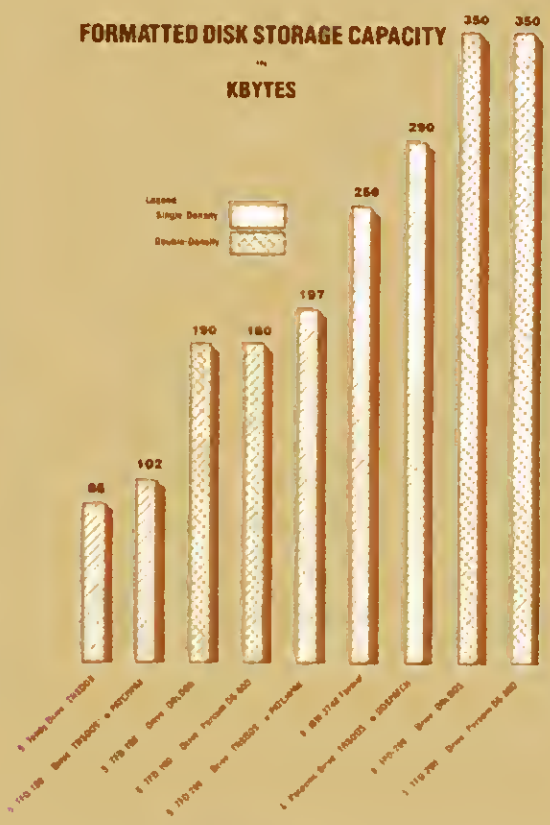
DVORAK VS. QWERTY:

The Super Keyboard That Never Caught On

74470 65947 12

TRS-80* Model I Computer Owners . . .

Store Up to 350 Kbytes on a 5" Disk



The DOUBLER™. It packs almost twice the data on a disk track as your single-density system. Depending on the type of drive, you can store up to **four** times more data on one side of a minidiskette than you can store using a standard Model I mini-disk drive.

- The DOUBLER™ reads, writes and formats **either** single- or double-density minidiskettes.
- Proprietary design allows you to continue to run TRSDOS*, NEWDOS‡, Percom OS-80™ or other single-density software **without making any changes** to software or hardware. Switch to double-density operation at any convenient time.
- Includes DBLDOS™, a **TRSDOS* compatible** double-density disk operating system.
- CONVERT utility, on DBLDOS™ minidiskette, converts files and programs from single- to double-density or double- to single-density.
- The DOUBLER™ circuit card **includes high performance data separator, write precompensation** circuits for reliable disk read operations — even with 80-track drives.
- **Plug-in Installation** — The DOUBLER simply plugs into the disk controller socket of your Expansion Interface, requiring no strapping or trace cutting. Expansion Interface disk controller may be completely restored to original configuration by simply removing the DOUBLER™ and re-installing the original disk controller chip.
- Works with standard 35-, 40-, 77- and 80-track mini-disk drives rated for double-density operation.
- Introductory price, including DBLDOS™ and format conversion utility on minidiskette, **only** $219.95. Use the coupon for even greater savings.

Mini-Disk Systems

More storage capacity, higher reliability — from Percom, the industry leader. One- two- and three-drive configurations in either 40 or 77-track format. Fully burned-in. From only $399

$20 PERCOM DISCOUNT COUPON $20
worth $20
toward
The Purchase of a
DOUBLER™
Coupon No. 80M103
Expires December 30, 1980
Void where prohibited by law.
$20 [illegible] $20

Call toll-free, 1-800-527-1592, for the address of your nearest authorized Percom dealer, or to order directly from Percom.

PERCOM

PERCOM DATA COMPANY, INC.
211 N. KIRBY GARLAND TEXAS 75042
(214) 272-3421

™ trademark of Percom Data Company, Inc
* trademark of Tandy Radio Shack Corporation which has no relationship to Percom Data Company.
‡ trademark of Apparat Company, Inc

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.
PRICES DO NOT INCLUDE SHIPPING AND HANDLING CHARGES

TRS-80* Model I Computer Owners . . .

Double-density storage. It's really here!

Here at Percom. And your authorized Percom dealers.

And double-density storage is here in a big way. Because now you can choose from *three different levels* of mini-disk systems — all *double-density rated.*

And get the storage that precisely meets your application needs.

Not to mention the service and quality that's made Percom the industry leader.

Although rated for double-density operation, all levels of Percom drives *work equally well in single-density* applications.

You can operate these drives in ordinary single-density format using TRSDOS*, Percom OS-80™ or any other single-density operating system.

Or, you can add a Percom DOUBLER™ to your Tandy Expansion Interface and store data and programs in *either* single- or double-density format.

Under double-density operation, you can store as much as *350 Kbytes* of formatted data — depending on the drive model — on one side of a five-inch minidiskette.

That's *four times* the capacity of standard Model I mini-disks, almost 100 Kbytes *more than* the capacity of the *eight-inch* IBM 3740 format!

Available in 1-, 2- and 3-drive configurations in all three model lines, Percom *burned-in, fully-tested* drives start at only $399.

TFD-40™ Drives

TFD-40 Drives store 180 Kbytes (double-density) or 102 Kbytes (single density) of **formatted** data on one side of a 40-track minidiskette. Although economically priced, TFD-40 drives receive the same full Percom quality control measures as TFD-100 and TFD-200 drives.

TFD-100™ Drives

TFD-100 drives are "flippy" drives. You store twice the data per minidiskette by using both sides of the disk. TFD-100 drives store 180 Kbytes (double-density) or 102 Kbytes (single-density) **per side.** Under double-density operation, you can store a 70-page document on one minidiskette.

TFD-200™ Drives

TFD-200 drives store 350 Kbytes (double-density) or 197 Kbytes (single-density) on one side of a minidiskette. By comparison, 3740-formatted eight-inch disks store only 256 Kbytes. Enormous on-line storage capacity in a 5" drive, plus proven Percom reliability. That's what you get in a TFD-200.

the DOUBLER™ — This proprietary adapter for the TRS-80* Model I computer packs approximately twice the data on a disk track.

Depending on the type of drive, you can store up to four times as much data — 350 Kbytes — on one side of a minidiskette as you can store using a Tandy standard Model I computer drive.

Easy to install, the DOUBLER merely plugs into the disk controller chip socket of your Expansion Interface. No rewiring. No trace cutting.

And because the DOUBLER reads, writes and formats *either* single- or double-density disks, you can continue to run all of your single-density software, then switch to double-density operation at any convenient time.

Included with the PC card adapter is a TRSDOS*-compatible double-density disk operating system, called DBLDOS™, plus a CONVERT utility that converts files and programs from single- to double-density or double- to single-density format.

Each DOUBLER also includes an on-card high-performance *data separator circuit* which ensures reliable disk read operation.

The DOUBLER works with standard 35-, 40-, 77- and 80-track drives rated for double-density operation.

Note: Opening the Expansion Interface to install the DOUBLER may void Tandy's limited 90-day warranty.

Drive enclosures, power supplies. Percom drive enclosures are finished in compatible silver enamel. Three sizes accommodate either 1, 2 or 3 drives. Drive power supplies are heavy duty, cool-running open-frame design. Three-wire ac power cords are safer, have lower noise pickup.

Free software patch. This software patch, called PATCH PAK™ upgrades TRSDOS* for operation with improved 40- and 77-track drives. For single-density operation only.

Quality Percom products are available at authorized dealers. Call *toll free 1-800-527-1592* for the address of your nearest dealer or to order directly from Percom. In Canada call 519-824-7041.

PERCOM

PERCOM DATA COMPANY, INC.
2[illegible] N. KIRBY • GARLAND TX • 75042
(214) 272-3421

Prices and specifications subject to change without notice. ☞ 258

™ trademark of Percom Data Company, Inc.

* trademark of Tandy Radio Shack Corporation which has no relationship to Percom Data Company.

80 Contents

PUBLISHER
Wayne Green

EXECUTIVE VICE PRESIDENT
Sherry Smythe

CORPORATE CONTROLLER
Charles Garniss, Jr.

ASSOCIATE PUBLISHER
Edward Ferman

ASSISTANT PUBLISHER
Jeff DeTray

ADVERTISING MANAGER
Kevin Rushelko

CIRCULATION MANAGER
(603) 924-7296
Debra Boudreau

BULK SALES MANAGER
Ginny Boudreau

ADVERTISING SALES
(603) 924-7138
Penny Brooks
John Gencarz

80 Microcomputing (ISSN -0199-6789) is published monthly by 1001001 Inc., 80 Pine St., Peterborough NH 03458. Phone: 603-924-3873. Subscription rates in U.S. are $18 for one year and $45 for three years. In Canada, $20—one year only, U.S. funds. Foreign subscriptions (surface mail), $28—one year only, U.S. funds. Foreign subscriptions (air mail), $60—one year only, U.S. funds. In Europe please contact Monika Nedela, Markstr. 3, D-7778 Markdorf, W. Germany. In South Africa contact *80 Microcomputing*, P.O. Box 782815, Sandton, South Africa 2146. Australian Distributor: Electronic Concepts, Attention: Rudi Hoess, 55 Clarence Street, Sidney 2000, Australia. All U.S. subscription correspondence should be addressed to *80 Microcomputing*, Subscription Department, P.O. Box 981, Farmingdale, NY 11737. Please include your address label with any correspondence. Postmaster: Send form 3579 to *80 Microcomputing*, Subscription Services, P.O. Box 981, Farmingdale, NY 11737.

PUBLISHER/EDITOR
Wayne Green

MANAGING EDITOR
Michael Comendul

TECHNICAL CONSULTANT
Jake Commander

PRODUCTION EDITOR
Clare McCarthy

ASSOCIATE EDITOR (COPY)
Carolyn Straub

NEWS EDITOR
Nancy Robertson

REVIEW EDITOR
Pamela Petrakos

ASST. TECHNICAL EDITOR
Chris Brown

EDITORIAL ASSISTANTS
Chris Crocker
Debra Marshall

TECHNICAL CONTRIBUTING EDITOR
Dennis Kitsz

EDITORIAL ADMINISTRATION
Cresca Clyne
Nancy Noyd

DESIGN ASSOCIATE
Diana Shonk

MANUFACTURING MANAGER
Noel Ray Self

PRODUCTION MANAGER
PUBLICATIONS
Nancy Salmon

ASST. PRODUCTION MANAGER
PUBLICATIONS
Michael Murphy

AD COORDINATOR
Sue Symonds

ADVERTISING PRODUCTION
Steve Baldwin, Bruce Hedin, Maryann Metivier, Dion Owens

PRODUCTION DEPT.
Joan Ahern, William Anderson Jr., Patty Burr, Linda Drew, Bob Dukette, Kenneth Jackson, Ross Kenyon, Patrice Scribner, Thomas Villeneuve

PHOTOGRAPHY
William Heydolph, Terrie Anderson, Bill Suttenfield

TYPESETTING
Barbara Latti, Sara Bedell, Michele Desrocher, Luann Keddy, Mary Kinzel, Linda Locke, Karen Podzycki

---

Cover artwork and illustrations on pages 66 & 67 by Alex Stevens. Alex is a brilliant, young illustrator attending Massachusetts College of Art.

MAKES EVERY BYTE COUNT

IN YOUR TRS-80™ MODEL I OR MODEL II DISK SYSTEM

NATIONAL HOSPITAL AND HEALTH CARE SERVICES, INC.

September 17, 1980

Dear Meta Technologies,

Because of my work load, this is the first opportunity tha I have had to write you concerning the programs that I have purchased from your company. The programs; CALCS, SHRINK and SIFTER have paid for themselves 1000 times over. I was able to take a custom written billing program which we had paid $2600 for and was able to condense it with SHRINK to about two-thirds of its original size! This was an incredible boon to my company as now I am able to fit several more utility programs on the same disk as the billing program. Just today I was able to adapt the 'SORTR' program in the series of sorts of SIFTER to work with our billing program. I believe that you understate the speed of this sort. In my experience, it is sorting over 500 records of 255 bytes in length in less than two seconds. As compared to the incredibly slow basic sort that I had in before, the 'SORTR' routine is just short of a miracle. Imagine having to wait over 45 minutes everytime a file of 500 records was accessed for sorting with the basic sort. If I had paid $500 for just this sort alone, it would have been worth it, as that is the amount of money it will save my company in the next six months. Now I have another eleven sorts in addition to the 'SORTR' program to adapt. This program, SIFTER, is worth many times what you are currently charging.

CALCS has outdone a series of programs(AIDS III AND MAPS) thatI didn't believe could get better. With the arithmetic manipulative qualities of CALCS I will be able to custom-write a total accounts payable/accounts receivable system. Not only that, but I am now able, using CALCS, to do sales, cost, and many other reports which require predicting arithmetically some future performance. Your program has completely revolutionized the paper-flow in my office. With the addition of NEWDOS+ I have an unbeatable software package. I can't thank you enough for the speed and error-free performance of your programs.

WHAT NEXT META-TECH ? How about revolutionizing the word-processing area ? You have an eager customer waiting to buy. I have yet to use your REM-ASSEM system because of my work load, but from the little I have done with it I am very satisfied. If you come out with anything new, please contact me.

Sincerely,

David E. Vareha, V.P. E.D.P. National Hospital and Health Care Services Inc.

2 Talcott Ave., Suite 36, Park Ridge, IL 60068, (312) 823-1617

Products damaged in transit will be exchanged. Prices, Specifications, and Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

DEALER INQUIRIES INVITED

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 5½% sales tax.

TO PLACE ORDER 1-800-321-3552

CALL TOLL FREE

FOR PRODUCT INFO 1-800-321-3640

IN OHIO call (216)289-7500 (COLLECT)

✓20

META TECHNOLOGIES CORPORATION

26111 Brush Avenue. Euclid. Ohio 44132

801022

TRS 80 is a TM of Tandy Corp.

1980 by Metatechnologies Corporation, Inc.

✓Reader Service—see page 274

80 REMARKS

"We're now renovating new editorial offices to accommodate the large staff that handles the publication."

The First Year of *80*

With your help, this first year of *80 Microcomputing* has turned out to be most successful. As far as I know, this is the first time any publisher has come out with a major magazine devoted to one single product.

It has been an interesting year for us. We're now renovating new editorial offices to accommodate the large staff that handles the publication. It takes a lot of work to put out over 200 pages of magazine a month.

Has any other technical magazine grown in less than one year to 125 pages of paid advertising? Certainly not in this field.

The aim of *80* has been to support the TRS-80 computer. The editors are under threat of serious personal damage to make sure that as much of the material in the magazine as possible is edited from computerese into English so that newcomers to the field will be able to come up to speed easily. Authors are warned that English is preferred to computer buzzwords. We'll leave the egotistical computer scientist baloney to others.

If you have developed any programs for the TRS-80, think of submitting them for publication. If you've come up with any accessories or ways to interface various peripherals to the TRS-80, the rest of us would like to know about it. Write it up. Articles are simple to submit; type them double spaced with generous margins and use upper and lowercase letters. The better the illustrations, the better the article. All articles are paid for upon acceptance, unlike some of the other publications.

My Short Editorials

Looking back over my editorials in *80*, I see that most of them have been relatively short. Mercifully short, perhaps. Older readers may remember John Campbell, who edited *Analog* for many years. Well, I was brought up on that magazine and I got used to the long, thought-provoking editorials John wrote, and so when I started my first monthly magazine, back in 1951, it never occurred to me to do anything but write long editorials.

John got to be a very good friend in the late 50s. There are few such brilliant men, so I was sorry to see him leave this world. But his influence lingers on in many ways, such as my editorials.

My writing reflects my many interests . . . heck, if they interest me, why not you? I write about trips, visits to manufacturers, to stores around the world, to shows, about skin diving, skiing, ham radio, Mensa, sports cars, CB radio, radar detectors, music, and anything else that happens to seem of interest at the moment.

I do quite a bit of writing in our *Microcomputing Industry* full of avuncular advice to the industry. Having called the turn of the coin several times with some accuracy, I occasionally get some respect, but not often. Move over Dangerfield.

I enjoy getting out and explaining to groups how they can make a success of their lives. It's so damned easy to be successful today, particularly in a field such as microcomputing where the whole industry is growing at an incredible rate. They and the country . . . and perhaps the world . . . would all benefit. Remember you don't bring benefits to a lot of others without benefitting yourself . . . and vice versa. This was the essential message of Adam Smith in the 1770s and it still holds true.

I also get annoyed when I run across people who are so damned docile that they will put up with bum laws just because they are laws. Have they forgotten that the Supreme Court has been batting down laws wholesale for years? Maybe one person can't fight city hall, but a group of us can. Sometimes I get into trouble over this, and sometimes I win. I took on the FCC a few years ago and helped to bring about massive changes in the ham regulations. The ARRL (American Radio Relay League) said it couldn't be done . . . don't try. I said to hell with them and pulled it off.

Well, I just wanted you to know a little about me, now that we've been together for 12 months. Perhaps you see why I light into Radio Shack every now and then. Sometimes I win a little. Sometimes not. They are almost as big as the government, and, in some ways, move about as fast.

Bad News for Software Houses

As Instant Software's distribution has grown, more and more small software producers have been asking about using our system to distribute their programs. Indeed, we've tried this with a couple of the more aggressive firms.

In the record industry, distribution has been consolidated by a few large firms. Smaller firms rely on them to distribute their records. The same thing has happened to magazine newsstand distribution; four or five large firms run the whole show. Perhaps we can learn from all that.

Instant Software has representatives covering the U.S. and 22 countries. Further, the size and number of programs Instant Software has in inventory enables us to talk directly to the major manufacturers of hardware and make a far better deal than could a small firm with a handful of programs. You can bet that we are out there selling the idea.

We've given some thought to helping other smaller firms with their distribution, but I wonder if this is really the best option for a small software house. Let me go into some details on that.

Firstly, if we were to do just the distribution, this would leave the duplication, documentation, packaging and advertising up to the software house. Yet in every step of that progression, a larger firm is able to keep its costs lower.

Buying blank cassettes or disks, for instance: Obviously sizable savings result from buying 100,000 or one million quantities. You can save even more if you do your own tape loading. But such machines cost $20,000 or so, putting it out of the reach of any but the largest houses.

Then comes the packaging. The design of a good package is expensive and has to be amortized over hundreds of thousands of packages. There is also the automatic machinery needed for putting the tape and documentation in the package and sealing it.

Small firms have to make do with poly bags . . . ugh. High volume packaging is much less expensive, again making it difficult for the small firm to compete.

META TECHNOLOGIES
FOR YOUR DISK SYSTEM

FILE BOX
DISKETTE STORAGE SYSTEM

$24.95 for 5¼" disks
for 8" disks . . . $29.95

MTC brings you the ULTIMATE diskette storage system, at an affordable price. Storing 50 to 60 diskettes, this durable, smoke-colored acrylic unit provides easy access through the use of index dividers and adjustable tabs. Unique lid design provides dust-free protection and doubles as a carrying handle.

PLASTIC LIBRARY CASES
(not shown)

An economical form of storage for 10 to 15 diskettes, and is suitable for your bookshelf! Case opens into a vertical holder for easy access.

5¼-inch diskette case $3.25
8-inch diskette case $3.50

Single Sided, Single Density, Soft-Sectored 5¼-inch, (for TRS-80™) Mini-floppy

DISKETTES
$19.80 ★ box of 10

Meta Technologies strikes again . . . at the competition! These are factory fresh, absolutely first quality (no seconds!) mini-floppies. They are complete with envelopes, labels and write-protect tabs in a shrink-wrapped box.

INTRODUCING
PLAIN JANE™
DISKETTES
The Beautiful Floppy
with the Magnetic Personality™

In 1980 alone, MTC has sold nearly a third of a million dollars worth of brand-name diskettes. If anyone knows quality, we do. And these are quality diskettes. The catch? They are in a plain white box. You're not paying for fancy printing, fancy labels or fancy names on the packaging. We don't even put our own label on the package (labels cost money). At this introductory price (our regular price will be $21.95 per box of 10) we cannot offer quantity or dealer discounts.

PLAIN JANE™ Diskettes . . . . . $19.80★

VERBATIM brand Diskettes (box of 10)

5¼-inch (for TRS-80™)
MD525-01 . . . . . $23.95
10 boxes of 10 . . . (each box) . . . $22.95

8-inch FLOPPIES
Single-Density, FD34-1000 . . . $29.95
Double-Density, FD34-8000 . . $39.95

TRS-80™ PRODUCTS

NEWDOS/80 by Apparat $149.95
NEWDOS+ with ALL UTILITIES
35-track $69.95
40-track $79.95
TRS-80™ DISK AND OTHER MYSTERIES . . . $19.95
MICROSOFT ™ BASIC DECODED & OTHER MYSTERIES for the TRS-80™ $29.95

META TECHNOLOGIES REDEFINES "Customer Satisfaction" IN 1981

met-a-mor-pho-sis (met'ə môr' fə sis), *n., pl.* -ses(-sēz'). 1. a transformation. 2. a change or successive changes in character or appearance. 3. **MTC's transition in 1981 to a bigger and better way of doing business, featuring new and improved products and services.**

MORE PRODUCTS ⟷

Products damaged in transit will be exchanged. Prices, Specifications, and Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

★PRICE GOOD THRU DECEMBER 31, 1980. Sorry, no dealer or quantity discounts. Allow for possible shipping delays.

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 5½% sales tax.

TO PLACE ORDER
1-800-321-3552

CALL TOLL FREE

FOR PRODUCT INFO
1-800-321-3640

IN OHIO call (216)289-7500 (COLLECT)

20

META TECHNOLOGIES CORPORATION
26111 Brush Avenue. Euclid. Ohio 44132

801010

Reader Service—see page 274

Documentation is a back breaker for small firms. Many solve the problem by having terrible instructions. A first rate product has to look good and have top-notch instructions. Even shipping is a problem for the smaller firm. To control its shipping, Instant Software *has* to warehouse all programs being distributed. We don't need the aggravation of having orders for something which is out of stock and over which we have little control. This means that smaller software houses will have to keep program packages in our warehouse, but at *their* expense.

Then comes the matter of advertising. This means the design and concept of the ad as well as its production. Once produced, the ad has to be run in the appropriate magazines. The more ad space you buy, the better the deal you get. Smaller firms pay a heavy premium for their small ads. And the worst part of this is that the larger the ads, the more response you get to them. There is just no justice... and no free lunch.

In every step a larger firm saves money. When you look at the situation closely, there isn't any step of the production and marketing of software that a larger firm can't do more economically than a smaller firm. If a smaller software house put more time into further programming, and left, not only distributing, but documentation, packaging and marketing to a larger firm, the net result should be far more real income. Instant Software, for example pays a royalty of 20 percent of its wholesale price to a programmer.

The day is not far off when some major firm is going to want several hundred Instant Software programs converted for their computer. One such bulk order can bring in a $15,000 royalty check just for the initial order—even for a simple game program! A more sophisticated business program may bring in an initial royalty of $40,000 or more.

You have me out there pushing hard for this type of sale for you.

Programmers who are already marketing their programs themselves have the option of submitting them to Instant Software, while continuing to sell the programs themselves. There's never been any problem with this. Instant Software asks only that there be no arrangements with any other third party software marketing firms. It takes several thousand dollars of investment before a new program package is ready to be marketed, so Instant Software wants to be sure that this is not going to be wasted by having the program come out from some third firm with a lower price. I hope that makes sense. ■

INSIDE 80

by Ed Juge, director of computer merchandising, Tandy Radio Shack

A couple of days ago, I received a phone call from a Model I owner. He was in the process of developing a rather unusual application to be used on a large number of TRS-80s. The Model I didn't have the disk storage he needed, yet he couldn't justify the cost of a Model II. Model III appeared to be a perfect solution.

Question: Would his already-written BASIC software work? Since...uh... mmm...his programmer had disassembled TRSDOS and was using some "undocumented" DOS routine....

My answer? Be prepared for a re-write! Radio Shack did it to another one... right? Wrong! The programmer did it to himself. Had he used only documented addresses, his program would have converted and run well on Model III. Addresses we don't publish, however, will change from one release of TRSDOS to the next (or in this case between Model I and III versions.) This is why we don't publish them!

The point of the story is—the programmer that has the savvy to pull such tricks (and there are lots of you), must have the foresight to anticipate the results. If the program is for your own use, no problem.

If you intend to sell the program to someone else... it could be a problem. Be sure it's understood what can happen if an attempt is made to use the program with a different release of either TRSDOS or our ROM.

Model III vs. Model I

Since I've indicated Model III TRSDOS is akin to Model I TRSDOS, let me explain. The ability to use Model I software on Model III does not mean that Model III is just a repackaged Model I! It is a new design with some intentional similarities. We tried to respond to many of your suggestions...those cards & letters do work!

We kept the 16x64 screen format for compatibility, with the same high-resolution monitor as built into Model II. We included the Model I cassette format so that the large base of Model I software can still be used. Yet at the same time we've included a new, faster, more reliable 1,500-baud analog cassette I/O system. We've given you a means of converting Model I disk software to Model III format, but Model III uses fast 40-track double-density drives.

Our popular 12K Level II has grown into a 14K Model III BASIC. Model III's BREAK key returns control to you from any operation, even LPRINT with no printer, or a bad CLOAD?, and you won't lose the resident program. Every key has auto-repeat, and there's a keyboard-controlled screen print feature. Model III also has a parallel printer port (so even Level I BASIC now has print commands). Model III BASIC includes the dual-speed cassette capability, a real-time clock, upper and lowercase drivers, a special graphics character set, and RS-232 I/O routines. You can define your own cursor character, blinking or non-blinking.

In your applications programs, you can protect up to seven lines at the top of the screen from scrolling during input to the screen. There's even a ROUTE command to direct specified outputs between keyboard, display, printer, and RS-232 (send or receive).

By the way, there are 24 pages of ROM addresses in the manual.

Model III TRSDOS

The Model III TRSDOS is more like the Model II than Model I. The directory and free space map are pure Model II, as are many of the features and commands. There are ERROR and HELP commands for quick reference, a fast string sort, and a variable cross reference utility. You can even write and protect a diskette via software command. M III's DO file capability allows a string of automatic keyboard entries which allows you to powerup in your application program after entering the date. A CONVERT utility allows Model I disk programs to be moved to a Model III diskette, without disturbing the Model I diskette.

A variety of commands provides for a number of functions. For example, CREATE allows pre-allocation of disk file space, DUAL duplicates output to video and printer, MASTER tells TRSDOS to always begin a disk search on a specified drive other than Drive 0.

There are also a series of CMD (x) commands, with different arguments serving a number of purposes. CMD(C), for exam-

META TECHNOLOGIES

FOR YOUR TRS-80™ DISK SYSTEM

PROGRAMMING TOOLS

TDAM $19.95
For Model II $29.95
Includes MTC QUE Card!

Having trouble with RANDOM FILES? With MTC's Table-Driven Access Method (TDAM) you'll never fret over FIELDing again. No knowledge of random access files is required. Insert the TDAM "interpreter" into any BASIC program and type in a few DATA statements describing the information in your files. TDAM does the rest! Reads and writes fields and records of any type (even compresses a DATE field into 3 bytes!). Features automatic file buffer allocation/deallocation, memory buffering, sub-record blocking/deblocking, and handles up to 255 fields per record. Super fast and super simple! Complete with TDAM interpreter, instructions and demo program. Requires programming experience.

SIFTER..... $19.95
For Model II..... $29.95

Twelve in-memory high-speed sorts for use in any BASIC program; stable, non-stable, with/without tags, for numeric or string data. Random File Sort included. Some sorts written in machine code. Includes sort subroutines, demo programs and instructions. Relocate as needed with REBUILD. Requires programming experience.

SHRINK..... $19.95
For Model II..... $29.95

Makes Every Byte Count! Make programs smaller and faster! Combines lines & removes unnecessary code including remarks, without altering program operation. Typically reduces program size 25% to 40%.

SUPERSEDE $19.95
For Model II..... $29.95

A "must have" for the professional programmer or the serious amateur. Probably one of the greatest time-savers available. Write programs in shorthand · change variable names · generate program documentation · use with REBUILD and MINGLE to build new programs from old ones.

MINGLE-II..... $19.95
For Model II..... $29.95

Merge up to 14 files (Program or Data) into a single file. Data files may be merged in ascending or descending sequence with the ordering based on a user-specified comparison field. A very handy utility for consolidating data files.

"OTHER MYSTERIES" VOLUME II

foreword by

H.C. PENNINGTON

Call now and place your order for this new book, "MICRDSOFT™ BASIC DECODED & OTHER MYSTERIES for the TRS-80™", from IJG, Inc. A primer for cassette and disk BASIC on the TRS-80™, the information provided applies to similar MICRDSOFT™ BASIC interpreters. Features include definition of terms, an overview of BASIC and DOS, explanation of exits, error codes, verb actions, "cold" and "warm" restart procedures and examination of system utilities, arithmetic support and I/O driver routines, and the communications region in RAM. Individual routines are explained in detail, with an index provided for easy access. Appendixes include tables for BASIC and DOS vectors, stacks and interrupt locations. PLUS thousands of comment lines for the complete MICRDSOFT™ BASIC.

MICROSOFT™ BASIC DECODED . . $29.95

The perfect supplement for your NEWDOS, from IJG, Inc.

"TRS-80™ DISK AND OTHER MYSTERIES"

by Harvard C. Pennington

132 pages written in PLAIN ENGLISH packed with HOW TO information with details, examples and in-depth explanations. Recover lost files and directories, remove file protection, make BASIC programs unlistable. How to use SUPERZAP, recover from DOS errors and MDRE!

TRS-80™ DISK..... $19.95

NEWDOS/80

by Apparat

Apparat's long-awaited successor to NEWDOS+ is here! This is not an enhanced version of NEWDOS, but a completely new product. Simplified DOS commands can be instantly executed from BASIC, even within a program, without disturbing the resident code. System options, such as password protection, number and type of disk drives, BREAK key enable/disable and lowercase modification recognition, can be quickly and easily changed. Five new random-access file types allow record lengths of up to 4096 bytes, and no FIELDing! A powertul CHAIN facility allows keyboard INPUTs to be read from a disk file. An improved RENUMBER facility permits groups of statements to be relocated within program code. Diskettes may even be designated as RUN-ONLY! Features all NEWDOS+ utilities (SUPERZAP 3.0, etc.) and much more! One MTC technical staff member said having NEWDOS/80 is "better than sex" (you'll have to judge for yourself!). Includes 180-page instruction manual and MTC QUE card.

NEWDOS/80..... $ 149.95

MTC QUE Card only $7.50

CALL REGARDING OUR NEWDOS+ UPGRADE PRICING.

Complete for Model I with all utilities
Plus exclusive MTC QUE card!

NEWDOS + $69.95

by Apparat

includes REF, RENUM, SUPERZAP, EDITOR/ASSEM., DISASSEM., DIRCHECK, and more! This is the original NEWDOS with all of Apparat's utility programs. Includes exclusive MTC QUE (Quick User Education) card.

MTC QUE Card only $1.50

MORE PRODUCTS

Products damaged in transit will be exchanged.
Prices, Specifications, and Offerings subject to change without notice.

MOST ORDERS SHIPPED WITHIN ONE BUSINESS DAY

DEALER INQUIRIES INVITED

WE ACCEPT
- VISA
- MASTER CHARGE
- CHECKS
- MONEY ORDERS
- C.O.D.

- Add $2.50 for standard UPS shipping & handling
- $2.00 EXTRA for C.O.D.
- Ohio residents add 5½% sales tax.

TO PLACE ORDER CALL TOLL FREE 1-800-321-3552

FOR PRODUCT INFO 1-800-321-3640

✓20 IN OHIO call (216)289-7500 (COLLECT)

META TECHNOLOGIES CORPORATION

26111 Brush Avenue, Euclid, Ohio 44132

801022

TRS-80 is a TM of Tandy Corp

✓ Reader Service—see page 274

ple, compresses program lines by removing remarks, spaces, or both (your option). CMD(L) loads a specified machine language routine to be called from BASIC, and CMD(O) sorts or alphabetizes the contents of an array to name just a few.

Of course, the physical differences reflect many of your suggestions, too. The one piece cabinet is capable of housing a Level I or II ROM, up to 48K RAM, RS-232, two of the four possible disk drives, and it's all fed by one power cable. To hold the non-disk price down, the add-on drive kit for drive 0 includes the controller and power supply for both internal drives. Again, the parallel printer port is built in, so we've eliminated the need for our ever-popular expansion interface with Model III.

You'll be happy to know that every effort has been made in Model III to eliminate Radio Frequency Interference. I hope it's obvious that Model III, both in hardware and software is anything but a "warmed-over Model I."

We've Been Getting Questions

Q: Can my Model I be retrofit to Model III specifications?

A: Unfortunately, no. The hardware is too different.

Q: Does the introduction of Model III mean that Model I will no longer be available?

A: As inevitable as death and taxes, the day will come when every product we make, even Model III and the pocket and color computers, will go. As of now, I honestly can't tell you when Model I will cease.

The real question behind this question is usually "Am I (Model I owner) now going to own a useless piece of outdated hardware? My answer is an emphatic *NO*! Discontinuence, when it comes, won't change the benefits of your computer or its value.

Q: Is it true that Radio Shack has been delivering 77-track double-density drives and just not telling anybody? If so, can Model I work with them?

A: All of our drives, from day one, have had double-density heads. Earlier this year, we began to use drive mechanisms which were capable of 40 tracks, not 77. Those are the drives we sell today. They are faster, band positioner drives, and work well with either Model I or III. Our engineers tell us that Model I would not work reliably in a double-density mode. By the way, the earlier 26-1160's and 61's won't work with Model III.

Q: Is there a modification to eliminate RFI for Model I, especially for ham radio use?

A: No. We don't know of a reliable way to eliminate RFI in existing Model I's. I've heard stories from people who have killed 90 percent of the interference, but they also tell me that others have tried their methods (some quite involved), without any real improvement.

Q: If I buy a Model III, can I be assured satisfactory operation with my ham equipment, without Radio Frequency Interference?

A: All computers generate some level of RFI and Model III like all of our computers, complies with FCC regulations. But it's our belief that the FCC tends to protect your neighbors—not you—from the computer. You may simply have to choose between computer operation and TV, ham radio, etc. I know RFI isn't a pleasant situation. I've spent a few nights on 20-meter CW with my unmodified Model I and M-80.

Visicalc Comes To TRS-80

We began shipping Model I Visicalc software back in late September. Although it was for Model I only, the plan was to have a Model III version available in November. On the chance that some of you are not familiar with Visicalc, it's worth explanation. You'll find it one of the most versatile programs.

> "Q: Can my Model I be retrofit to Model III specifications? A. Unfortunately, no . . ."

Picture a large spread sheet with up to 63 vertical columns labeled A, B, C, etc., and up to 256 horizontal rows labeled one, two, three . . . and so on. (There are a few restrictions depending on memory limits.) Visicalc turns your computer's memory into such a sheet. Now, any location on the sheet (A1, AK211, O7, etc.) can contain a label or heading, or a number, or a formula telling the computer how to calculate the figure there. For example, A32 could be Net Profit, and could be A6 (Gross Profit) minus A30 (Total Expenses).

Now, the fun. If you went to do income projections for a year, enter your first month, then tell Visicalc to assume a 5 percent monthly increase in sales. You can handle fixed expense items by repeating those exact numbers for all 12 months. And when you're finished, when you enter or change a figure, Visicalc performs a bit of magic. The new numbers are calculated and put into place immediately. Adding other variables is quick and easy. For example, what if you add a new employee, buy a new delivery truck, move into a less expensive office, start an expensive ad campaign, increase sales faster, have a sales slowdown?

Your bills can be adjusted for seasonal variations by entering a fixed amount or a relationship to a base month.

Your video screen acts as a window on your spread sheet, and you can move the window anywhere. You can lock headings in place while scrolling, or even split the screen horizontally or vertically, scrolling only a portion of the display. Of course Visicalc can print specified portions of your spread sheet, or store it all on disk for later use. The possible uses are endless.

Whether you're calculating the family budget, or doing corporate financial planning—if you own a TRS-80, Visicalc would be an excellent addition. (Yes, Tandy's financial wizards have been using it for some time, too.) And yes again, it will be available for the Model II in December if Murphy doesn't butt in.

The Management Computer

With the introduction of Visicalc, Scripsit, TRS-80 Videotex, and Profile, and especially with one-piece hardware like the Models II and III, businesses are finding that the TRS-80 is a valuable management tool. It saves time and labor, manages date, does financial planning, and even composes memos or letters. It also can provide a low-cost electronic mail service by means of Videotex software and the CompuServe Information Service. And best of all, it's not just another future concept. It's all happening today.

If you haven't tried the CompuServe, you should. Our TRS-80 software packages include a free hour on the system. Your unique user number and password come right in the software package. There is no sign-up fee even if you decide to keep going after your free hour. Whether or not you find a home with CompuServe, the Videotex gives you an excellent general purpose communications package.

Visicalc, Scripsit, Profile and Videotex all make outstanding gifts for computer owners, and are almost universally usable regardless of the first use of the computer.

Also remember that cassettes, diskettes, and dust covers make outstanding low-cost gifts. But for you we at Radio Shack want to wish you, a happy holiday season and an outstanding 1981. We hope your stocking on Christmas morning comes stuffed with outstanding and unique computer gifts. ■

80 INPUT

"He was a first-class communicator . . . I for one will not forget his discussion of the appetite for the miraculous in modern man."

Micro Millenium

When reviewing *The Micro Millenium* for your August issue, Nancy Robertson was clearly unaware that the author died suddenly before the book was published. This may well explain the deterioration she detects in the later chapters.

In addition to his qualifications as a psychologist, Christopher Evans was an applied computer scientist in the British 'boffin' tradition. His pioneering work on dialogs between microcomputers and naive users led to the development of the MICKEY system, in which an unattended microcomputer is used to gather medical information from patients. (On some sensitive topics, it seems that patients will give more accurate information to a sympathetic microcomputer than to a human doctor, and prefer to do so.)

He was also a first-class communicator of scientific developments to the non-scientist, combining a flair for exposition with sturdy common sense. I for one will not forget his discussion of the appetite for the miraculous in modern man, culminating in a demonstration of spoon-bending as performed by Uri Geller, given in swimming trunks during the lunch break in a symposium on 'Man-machine interaction' in Greece.

His was a rare combination of talents, and his death has left a gap that will not easily be filled.

Dr. Hugh David
91830 Le Coudray-Montceaux
France

A Ham Writes

I have never written to you before although as a Ham I have read *73 Magazine* for a number of years and now subscribe to *80 Microcomputing*. In passing, I must say that I admire your style. You are one of my favorite curmudgeons. I use the term affectionately since I aspire to the same estate.

The reason for the letter is to tell you that the second issue of *80 Microcomputing* I received more than paid for several years' subscription. I was just about to disconnect my interface to drive to Nashville (35 miles) because of a problem that had developed between my TRS-80 and my H-14 printer. Both seemed to be operating well but the information the computer was sending was printing out as garbage on the printer. The October issue came just as I was about to start pulling things apart, so naturally reading the magazine took precedence over fixing the computer. The issue fell open to James Kunzman's article about the NEC Spinwriter and his problems with a warped RS-232-C board. To make a long story shorter: with a screw driver, a pair of tweezers, and a pencil eraser, I was able to fix my problem without a $24 trip (plus mileage) to the Radio Shack repair center.

I hope you and your magazine live forever.

Donald R. Goss, Chairman
Division of Humanities
Gallatin, TN

Bat Guano

I just received my fourth issue of *80 Microcomputing*. The house rule here is that three back issues of any magazine remain; all others are tossed out. But *80 Microcomputing* seems to contain so much information that I can't use it all at once but must keep the magazines on file for a time when I can get around to it all.

After four issues, the quality of the information I'm keeping is coming strongly into doubt.

From the June issue, I patiently keyed in the "Life" listing. I realized in the process that something was wrong with the program, worked around it, and saw the corrections in the July issue.

In the August issue, I patiently keyed in the "Swords and Sorcery" program, again realizing that that program was screwed up, too. You can jimmy a missing symbol in an assembly program, but it's hard to supply missing data in a BASIC program. These missing items apparently had nothing to do with the programs themselves, but were just results of shoddy editing.

This month—September—I tried "Divine Proportions," and found that the programming itself was downright shoddy—the parallelepiped, for example, was drawn wrong, and the option to print out the proportions clobbered the screen display. Again, if *80 Microcomputing*'s staff had checked out the program, this shouldn't have happened. Another case of editorial irresponsibility.

80 Microcomputing also carries hardware modifications. I haven't tried any yet, but I have the fear that if I did, I might wind up with a smoking wreck instead of a microcomputer if the hardware modifications are as shoddily checked as the software that's published. Ol' Madman Wayno can rip us off only so far.

1001001? 101101—bat guano on your antenna.

Richard Amyx
San Jose, CA

Richard, see previous letter.—Eds.

Article Argument

Your August 1980 issue quoted and snidely commented on an article in the June 16, 1980 issue of *Business Week* wherein two Tandy VPs stated flatly that no new computers were coming out this year.

When I queried Radio Shack about the item, they reported an inability to find the quoted article in that *Business Week*. Nor could I. When I brought this to your attention, I was directed to the June 9, 1980 issue of *Business Week*. But I came up dry there as well. No interview, no article.

It looks from here that an apology is due the Tandy people. At the least, it is no way of healing the breach between your magazine and their advertising department.

John R. McGinley, Jr.
New York, NY

I suggest you read a little more closely before sending off your next letter to an editor. The first three paragraphs of the 80 piece you referred to are reprinted here.

"An article printed in Business Week,

June 16 stated that, "Over the next six weeks Tandy plans a barrage of new products to follow up its initial foray into the small business market with its TRS-80 Model II."

"It goes on to say that a desktop computer for scientists and engineers, a word processor based on the Model II and small computers that will automate inventory controls are to be expected.

"At Tandy Corp., both H. L. Seigel, National Publicity and Promotion Manager, and Senior Vice President of Operations, Charles Phillips, deny the thrust of the Business Week *article. They both say no new computers that they know of will be marketed by the company before the end of the year."*

It is not the Business Week *article that quotes the two Tandy executives who denied knowledge of the new computers, it is* 80 *that quotes them.* Business Week *had the scoop. We tried to follow it up, without immediate success. The September cover story gave the details that would have been nice to have had in August.*

The Business Week *article is indeed in the June 16 issue on page 106J, entitled "Tandy's personal-computer salvo."*

Nancy Robertson
80 News Editor

Dislikes Content

I'm sorry, but the time has come when I feel compelled to write regarding your current editorial content regarding "software pirates". Believe me, I do not condone this practice, but I feel you are using your publication, supported by my subscription fees, to promote a very self-serving, vested interest on your part as head of Instant Software.

I subscribed to *80 Microcomputing* because it was advertised as being the top of the line in publications dealing with the TRS-80, not because it was to degenerate into a soapbox for the slanted views of its publisher or to become saturated with advertising material.

As examples of the current state of your publication I cite the following:

In the last issue of *80 Microcomputing* there were fourteen pages devoted to the software pirating on the copyright situation and one hundred twenty-five pages of advertising meterial.

These examples amount to the following percentages:

Pirate to TRS-80 material	6.2% of total contents
Advertising to TRS-80 materiel	55.3% of total contents
Total editorial/advertising	61.5% of total contents

As a businessman I realize advertising revenue and editorial content are important parts of any publication, but after discounting some questionable material and still arriving at a total subscriber oriented magazine content of 38.5% of the 226 pages of your last issue, I honestly think you are putting the cart before the horse at the expense of your readers.

I for one would like to see you and your publisher friends get off your collective soapboxes and get back to the business at hand—publishing some top quality magazines dedicated to the top selling computer in the world—the TRS-80.

Vern H. Hall

Though we can't agree with your breakdown of editorial matter and are confused at your reasoning concerning articles covering software copyright or piracy—Wayne's editorials aside—we would like a chance to respond to criticism that our magazine is becoming more crowded with advertising.

Nowhere has the pressure to introduce more editorial matter in the magazine been more sharply felt than in our own offices. This pressure has not only come from upper management, but from our own sales department.

The editors are in complete agreement with these sentiments.

Perhaps, to a reader, our magazine's growth rate and the problems encountered are not self-evident. A publication which appeared in January 1980 at 147 pages and appears just one year later at over 250 has undergone radical changes. The editors have been chasing advertising space sales since February.

The more hectically we operate, the more difficult it is to guarantee both lucid and accurate articles; articles which we hope depart from what passes as "technical" literature to become both educational and even enjoyable.

Only now are we properly staffed to do the necessary job. In the future we hope to offer yet a larger and more carefully edited magazine.—Eds.

Connector Advice

Radio Shack now sells a 40-pin card edge connector (Part 276-1558, on page 126 of their 1981 catalog) which they describe as "compatible with many microcomputers." While this connector fits the TRS-80 expansion port, from the back, the TRS-80 has the low numbers at the left, and the odd numbers on top of the printed circuit card. With the card edge connector held so that the low numbers are at the left, the even numbers are at the top. This causes no problem if two card edge connectors are used on a single cable, but anyone who uses the connector to attach the TRS-80 to a circuit board should be aware that the wires will be ordered 2, 1, 4, 3,. . . . .38, 37, 40, 39 and not in the usual sequence.

Sherman Levine
White Plains, NY

Scripsit: Round ?

Further to the letter in your September issue from William O'Brien: I have now been using Scripsit for about three months, and am generally happy with it, but I am still bugged by the lack of line feed with the Enter key. I've become used to it and work around it, but it would be nice if it could be made to work.

I use my TRS-80 with an IBM Selectric typewriter and Escon (USA) interface unit. With this, the Enter key doesn't give a carriage return unless there are several characters in the line. This means that such things as block formats and paragraph formats don't work.

You could say that I should have used a Tandy printer, but I won't buy that! For a start, in an office environment you must be able to produce typewriter quality printouts, and Tandy had not released their daisy-wheel printer here at the time of writing. They have no other printer with the needed quality.

Also, for business correspondence you must have a number of other features, such as half line spacing, both for setting out and so you can write things like CO_2 or MC^2. You *must* be able to underline and you should be able to use all of the characters on the type ball, including ¼ and ½. It is also nice to be able to correct small mistakes in the middle of the printed page, which you didn't see until you printed it out, using the typewriter's corrector ribbon. No printer can give you that with Scripsit—except the Selectric.

To get all of this you need either a dedicated word processor (at $15,000+) or a printer which doubles as a typewriter. This is what I have, and if I could just get it to execute a carriage return with the Enter key, I'd be happy.

Hope someone out there has had the same problem and solved it!

David D. Harris
470 Manon Rd.
Plympton Park
S. A. 5038
Australia

80 AID

Needs Equipment

As you probably know, the Coast Guard Auxiliary is a nonprofit, volunteer organization whose primary purpose is to assist the Coast Guard. Here in the Caribbean, our group operates mainly in the area of Search and Rescue.

We have a considerable amount of information to assist us during SAR operations that is presently stored in files, books, boxes and on scraps of paper. When time is of the essence, we must rummage through all of this material to find the needed information. The suggestion has been made that all of this data could be stored in a small computer.

Our problem, however, is that we do not have any government funding and all of our equipment is purchased by individual members. The purchase of a microcomputer would be out of the range of our people since most of us are retired and live on fixed income.

Is there any possibility of one of your readers donating a used or out-dated computer to our group? As a nonprofit organization, we are permitted to accept donations of this type and the donor would receive a favorable tax write-off.

It would be greatly appreciated if you could assist us in this matter.

Milton Greene - Vice Commander
Coast Guard Auxiliary St. Croix
Box 2759 - Christiansted
St. Croix - USVI - 00820

Bi-Sync *v.* A-Sync

I have what appears to be an unsolvable problem which might be of interest to both you and my fellow readers and just possibly of enough potential to engender the development of a solution!

On the surface the situation seems simple: I operate a 32K TRS-80 with disk drives. With the R 232 board and a 300 baud R/S, modem communications with the outside world are a delight.

My corporate headquarters, on the other hand, operates a monster IBM System 34, communicating with the world in 3741 protocol via 201C 2400 baud dial-up equipment.

Trouble is that the system 34 communicates in Bi-Sync. I need to communicate in A-Sync.

Short of spending upward of $5,000, there appears to be no solution to my problem. Software packages are not available. Conversations with data communications companies who might act as an interface have proven either fruitless or way too expensive.

Any ideas??

Raymond L. Watkins
ICC Industries, Inc.
Dover Chemical Corp.
Davis at 15th St.
P.O. Box 40
Dover, OH 44622

Scripsit Reboot Aid

Reference *80 Microcomputing* for the month of July; there was a gentleman concerned with rebooting to DOS when using SCRIPSIT. Have no fear, if your system does reboot all you have to do is go to DEBUG and execute G6008. SCRIPSIT will come back with no loss of data. Do not try going to BASIC after reboot and enter by SYSTEM: Your data will be lost for sure.

There is also another answer to reading the directory when in SCRIPSIT, and that is to buy NEWDOS 80 by APPARAT Inc. of Denver, Colorado. This is an excellent DOS system and allows you to read the directory without losing SCRIPSIT.

Chuck Gould
Route 6 Box 6460
Nampa, ID 83651

Keyboard Bounce

I have a problem; but first a word of explanation.

I am a graduate mechanical engineer. I own a TRS-80, Model II with 16 K. This was originally a Model I, 4K which I have had modified (by Radio Shack) to Level II, 16K.

Each morning that I intend to use the computer I first have to CLOAD a cassette titled Keyboard Debounce, Systems * KBFIX.

Without first loading this cassette I have to go through the throes of keyboard bounce (multiple printing).

I can't understand why loading this KBFIX cassette will solve the keyboard bounce, but the repair shop tells me that the affliction (to the computer) *cannot* be repaired on a permanent basis internally.

Granted, loading this cassette consumes only a few minutes time and the computer performs beautifully after loading it. But I'm 63 years old and I like to put my time in more beneficially.

John V. Lane
14400 Astoria Street
Sylmar, CA 91342

DATEL Aid

I recently purchased a DATEL 30 Selectric based I/O terminal. I have been unable to find documentation and DATEL is now defunct. The terminal is EBCDIC encoded and I would like to convert it to ASCII. Any help in this conversion or documentation on the electromagnet driver board, power supply board, logic board, or the 50 pin connector between the logic and the typewriter would be greatly appreciated.

Brad M. Dickey
2806 Treehouse Pkwy
Norcross, GA 30093

Needs Interface

My surplus Datel Selectric came with a software driver that works fine on programs, but it will not work with a word processor. I have tried both Pencil and Scripsit, and it will not print. Can anyone suggest an interface?

Paul Kalkstein
Phillips Academy
Andover, MA 01810

How's your love life?

A little dull around the edges? Routine? Predictable? Boring? Maybe all it needs is a little Interlude. Interlude is the most stimulating computer game ever conceived. It combines a computer interview, an innovative programming concept and a one-of-a kind manual to turn your love life into exciting, adventurous, delicious fun!

Interlude is: romantic ...playful ...outrageous ...a fantasy. Interlude is: ■ A Bed of Roses (Interlude #1) ■ Mata Hari (Interlude #49) ■ The Chase (Interlude #7) ■ Rodeo! (Interlude #71) ■ The King and I (Interlude #60) ■ Some Enchanted Evening (Interlude #84) ■ Caveman Caper (Interlude #82) ■ From Here to Ecstasy (Interlude No. 30) ■ Satin Dreams (Interlude #72).

More than 100 Interludes are included in the program. Most are described in detail in the accompanying manual, but several surprise Interludes are buried in the program awaiting that very special time when your interview says you're ready. (When you learn secret Interlude #99, your love life may never again be the same!) Interlude can give you experiences you'll never forget. Are you ready for it?

Interlude™
The Ultimate Experience.

287

INTERLUDE, 10428 Westpark, Houston, Texas 77042. I'm really ready. Send my Interlude today.

Apple II (16K)*
☐ Cassette ($16.95)
☐ Diskette ($19.95)
☐ Diskette—Pascal or DOS 3.3 ($19.95)

TRS-80 (Level II 16K)**
☐ Cassette ($16.95)
☐ Diskette ($19.95)

Poster
☐ 20"x 24" reproduction of this ad without ad copy ($4.95 includes shipping charges)

Add $1.50 for shipping and handling

Available for immediate shipment.

Please enclose your check payable to INTERLUDE or complete the charge information:

☐ MASTERCARD ☐ VISA All charge customers must sign here.

Account No. Expiration date MasterCard Bank Code

CHARGE CUSTOMERS. Order by phone toll-free! **1-800-231-5768 Ext. 306** (Texas 1-800-392-2348 Ext. 306)

Name Age

Address

City State Zip

*Apple II is a registered trademark of Apple Computers, Inc. **TRS-80 is a registered trademark of Radio Shack, a Tandy Co.

Book Review

TRS-80 Disk & Other Mysteries by H. C. Pennington is an exciting title isn't it? Here, I thought, is a book that will tell all about how the disk knows where to start and when to stop, how the system knows where all the pieces of a fragmented program are located, how a multiple disk system knows which drive to use. Perhaps it will give me hints on changing code on the disk so I can change the start-up process. This book promises to answer all my TRS-80 disk related questions.

The book starts out nicely enough; there is a short paragraph that says the TRS-80 is a pretty neat machine and Pennington loves it. Then there is a page about the crumbs in Fort Worth who are responsible for the existence of the TRS-80. We learn that the stupes who put TRSDOS together obviously didn't know what they were doing, but there are a pair of heros in Colorado who have saved the day for all us TRS-80 disk owners.

Most of the rest of this book tells us about the mistakes in Radio Shack's TRSDOS and the inadequacies of some of the other disk systems and how great NEWDOS is. We learn that NEWDOS works and fixes all the mistakes. The book then describes how to use NEWDOS commands and features. The conclusion I drew is that the documentation with NEWDOS is inadequate and I have spent $22.50 for an instruction book that I don't need. The title should have been *Newdos & Other Mysteries*.

John Grass
Portola Valley, CA

Sinclair Slips

After reading Mr. Sinclair's excellent article, "Into The 80's", I have found the following bugs:

1) The power switch does not perform a memory clear function during power-up. This is performed by the combination of a inverted input NOR gate, (Z53 & Z52), & the RC network, (R47 & C42). See R/S technical reference manual for details. Three poles of the four pole switch are used to switch the three power supply connections, (+5 & −5 vdc & vac).

2) When a reboot occurs, "Memory Size?" appears during program execution and you do not always lose the program in RAM. When it happens press the reset button *first*, then pull a list to see where your program went wrong. (By the way, the explanation given by R/S of the function of the reset button is incorrect (L II manual pg. 1/2). Pressing reset returns the computer to "READY", not to "Memory Size?". Also you do not lose the program in RAM.)

3) The maximum characters per line allowed is 255, not 250.

4) On Print@ syntax, the comma should be used after the position argument, (PRINT@64,"..."). This mistake was in the article, example 4 is correct.

5) This is not a bug but a method I have used to rid my keyboard of bbounce. Clean the contacts as R/S has recommended. Then paint the contacts with "Blue Stuff" or a similar product. Spray a small amount of the chemical into a paper cup, then dip a flat toothpick into the foam and *gently* paint the contacts. The chemical is a mild polishing formula which wipes the contacts each time they close. This chemical is used on TV tuners to keep the gold contacts clean.

John F. Costello
Philadelphia, PA

POKE Convert

It seems that both Bertram Thiel ("Double Size Graphics", June *80*) and Jeff Elsen (Input column, September *80*) have neglected to mention that there are more ways to escape the 32 character mode than CHR$(28) and CLS. I have found that POKEing 0 into memory location 16445 will effectively convert the video contents back to 64 characters per line and will leave the cursor where it is while CHR$(28) brings it to position (0,0). If for some reason you don't want to use CHR$(23) to enter the wide letter mode, POKE 16445,8 will do the same.

Benjamin Junge
Los Angeles, CA

Dancing Numbers Program

```
>LIST
5  CLS
10 B = 1
20 FOR X = 0 TO 895 STEP B
30 PRINT@X, B
40 NEXT X
50 B = B + 1
60 FOR L = 0 TO 75 : NEXT L
70 IF B > 891 STOP
80 GOTO 20
READY
>_
```

Try it just for fun!

I also would appreciate hearing from readers with programs helpful to the blind.

John Rego
Rt 2 Box 19
Logan, IA 51546

Continued to page 28

80 DEBUg

Math Flash Bugs

Corrected Lines for "Math Flash", Page 158, Sept 1980, *80 Microcomputing*:

```
95  ON Y GOSUB 1000,1100,1200,1300,1400,1500,1600,
    1700,1800
100 IF Y = 0 THEN GOSUB 1800
105 ON D GOSUB 2000,2100:W = 0:Y = 0
220 IF G C THEN PRINT @ 0,G;" IS WRONG. etc (Rest
    of line remains unchanged).
```

Jim Barbarello
R.D. #1, Box 241N
Tennent Rd.
Englishtown, NJ

Machine Language Bug

Got a friendly call from Nashville Tenn, this PM from a Ham who was trying to make sense out of an article of mine in the August *80 Microcomputing*, "Towards Mechine Language". There was a foul-up in the printing on page 144. Under the heading Mechine Code Listing, using T-BUG, punch in this series of commands starting at memory location 5000:

```
CD F6 04 3E 31 32 20 3E 76
```

The only nice part about composing room errors is that I find great numbers of folks out there who appreciate the effort that goes with authorship.

Allen S. Joffe W3KM
1005 Twining Road
Dresher, PA 19025

EXCITING NEWS FOR INVESTORS WITH TRS-80 32K DISK-BASED SYSTEMS!

Standard & Poor's proudly announces STOCKPAK, a unique software and data system to help you meet your investment goals like a Wall Street professional.

STOCKPAK not only delivers a "stand-alone" Portfolio Management System but also gives you the software for Standard & Poor's monthly Common Stock Data Service (available to TRS-80 owners on a subscription basis). With STOCKPAK and the Data Service you command one of the most powerful and versatile investment tools available.

Here's How STOCKPAK Will Help You:

A 900 COMPANY DATA BASE SERVICE

Monthly Data Service subscribers receive a diskette containing 30 vital financial items on 900 of the most widely traded stocks (S&P "500" and 400 NYSE, ASE and OTC issues). Accompanying this monthly diskette is an Investor's Newsletter highlighting important financial news and investment strategies, with suggestions for maximizing the usefulness of the system.

STOCKPAK SELECTION SYSTEM

The heart of STOCKPAK is a powerful, analytical stock selection tool which enables investors to choose stocks which meet their investment criteria. For example, you may wish to select only those oil and gas stocks with price/earnings ratios of less than 7 and yields of 6% or more. Once a group of stocks has been selected, you can store it as a separate data file for continuing use.

REPORT WRITER

You can define the report formats you would like to see on those stocks meeting your investment objectives. Hundreds of calculations and ratios that you define can be sorted, averaged or totalled, and displayed on video screen or optional printer.

PORTFOLIO MANAGEMENT SYSTEM

Now you can effectively evaluate and manage your own stock portfolio of up to 100 securities with as many as 30 transactions for each. You can record "buy" and "sell" transactions, price and dividend information and stock splits for instant retrieval, for record keeping and tax purposes. You can measure actual performance or create hypothetical situations to help you make "buy" or "sell" decisions.

HOW TO ORDER STOCKPAK

STOCKPAK is designed exclusively for TRS-80 users with 32K business systems with two mini-disk drives. You can obtain the basic software and sample Data Base, plus a comprehensive User's Manual from your local Radio Shack Store for only $49.95. The STOCKPAK Monthly Data Updating Service can be ordered directly from Standard & Poor's for $200 annually, or from the order form provided in the basic package you purchase from Radio Shack.

Standard & Poor's Corporation

25 BROADWAY, NEW YORK, NY 10004 (212) 248-3994/3374

Reader Service — see page 274

154

80 ACCOUNTANT

by Michael Tannenbaum C.P.A.

"While this may seem to be a lot of Mickey Mouse work, it is really sound data processing practice."

Recently I demonstrated the Radio Shack Payroll System for a client. After entering the data for several new employees, the program halted with a cryptic message OV ERROR IN LINE 1132. Covering my confusion with a humorous remark about Murphy's Laws, I listed the line.

The line contained a multipurpose routine used for both alphanumeric and numeric input, that determines the value of the input string. Unfortunately the input data contained an address 229 E 69th St. which the program had interpreted as 229 raised to the 69th power.

Of course I was able to make a quick fix (E became East) but the experience was quite distressing. If an inexperienced clerk had been operating the computer, he might have quit in disgust. Since the error occurred with the payroll master file open, the file could have been destroyed.

This type of bug is quite difficult to foresee. I am sure that future programs will correct this oversight and have procedures to facilitate an abort in case of an incorrigible error. However, this experience provides an important lesson.

A New Product

The Payroll System is a new product. New data processing products, both hardware and software, are prone to strange and unanticipated errors. My experience has been that most new systems of any complexity require at least six months of field operation to purge bugs. For this reason an older software product supported by a reliable software house often offers the safest path to reliable automation of your business recordkeeping system.

One of the first microcomputer accounting packages was the Osborne System. This system was developed by Adam Osborne and Associates for the Wang 2200 in the mid '70s. Originally written in Wang extended BASIC, it has been converted by many vendors for both the Model I and Model II.

The system has been thoroughly field tested and documented in a series of published manuals. In the latest manuals, the Wang BASIC listing has been replaced by CBASIC listings. Many reputable software houses offer versions that are quite low in cost.

The version that I tested was obtained from the Small Business Systems Group, Main St. and Lowell Rd., Dunstable, MA 01827. They have chosen to offer the system as a series of stand alone modules (accounts payable, general ledger, accounts receivable and payroll) or as an integrated Accounting Recordkeeping System.

Either method of application offers some advantages. In an integrated system each subsidiary recordkeeping module contains a program which prepares data for entry into the General Ledger program. At the end of each month, a special program generates general ledger information, eliminating hand journal entries.

Technically, an integrated system can eliminate close out journal entries that are required each month. This would increase accounting accuracy many times. By posting the recurring journal entries (for example depreciation, cost of sales and amortization) and financial reports can be prepared automatically.

Some Sacrifice

Alas, nothing is obtained without sacrifice. The catch is that an integrated system must sacrifice disk capacity to contain all the programs and data files on one set of disks. The integrated system for the Model II only accommodates 400 receivable customers and 400 payable vendors. The General Ledger was limited to 200 accounts.

There is also a limit on the number of open transactions which can remain in the system. All limits can be doubled by using an additional disk. Fortunately receivable capacity—or the capacity of any other module— can be expanded at the expense of other modules. The Small Business Systems Group (SBSG) thoughtfully included the variable designations for file limits used in each subsystem.

With 34 programs and 13 files the integrated accounting system represents an outstanding value. All major functions are menu driven and it will be easily learned. In addition, an invoicing module is included to automate billing operations.

The package is supplied on two dual density eight-inch diskettes with a 24-page description of the system and directions. Buyers are clearly directed to purchase the Osborne manuals. The system description is not intended to provide the detailed information that is available in the manuals.

All menus and functions are as specified in the Osborne manuals with the exception of the invoice module and the separation of the accounts payable and receivable main menus. All edit checking and data limit testing specified in the manuals are included. However, the job cost provisions of the original system have been eliminated.

The one new feature, invoicing, greatly extends the usefulness of the package. Designed as are the other modules, invoice data is entered into a transaction file, where it can be altered by a file maintenance procedure. When all data is correct, an invoice printing routine is selected. An additional routine prints shipping labels.

Limited Capacity

The capacity of the transactions file is limited to 50 items. To purge it, the Accounts Receivable Update program must be run. It transfers the invoice totals to the Accounts Receivable transaction file, which does not update the receivable records directly. To accomplish this task the Accounts Receivable Update program itself must be run.

While this may seem to be a lot of Mickey Mouse work, it is really sound data processing practice. No file is updated directly in the Osborne System. All files are batch updated with hard copy control totals generated for each batch. This provides an independent audit trail which should be used to control the accuracy of the data retained in the system.

Unfortunately the use of a batch update procedure creates a potentially dangerous situation. In a batch system entered data is usually subject to adjustment. Entries should be pretotaled and totals balanced to the computer batch proof totals after entry. If the totals are out of balance, an adjustment can be made.

All modules in the Osborne system can

Give something different this season — the pleasing gift of increased memory — to your favorite TRS-80* user. The MT-32 from Microtek. The new, brilliantly designed Printer/Memory expansion module for the TRS-80 Model 1.

This unit will add 16K or 32K of RAM to the basic 16K machine without the expense of a full blown expansion interface. The module also contains circuitry to drive Microtek's MT-80P dot matrix printer or any other Cantronics-compatible printer. No special software routines. No hardware modifications. Attaching or detaching takes seconds. One year warranty.

Three configurations are available:

- Without RAM assembled and tested (MT-32A @ $119.50)
- With 16K RAM assembled and tested (MT-32B @ $159.50)
- With 32K RAM assembled and tested (MT-32C @ $199.50)

Available from Microtek or your nearest computer dealer.

MICROTEK inc.

9514 Chesapeake Drive
San Diego, CA 92123
Tel. (714) 278-0633
TWX 910-335-1269
Outside California
call toll free: 800-854-1081

360

* TRS-80 is the Registered Trademark of Radio Shack, Div. Tandy Corp.

adjust unentered transaction batches. It is quite possible, therefore, to transfer an invoice to the Accounts Receivable module and delete it or modify it before posting.

To guerd against this situation, the invoicing module provides a series of hard copy reports which are genereted when data is transferred.

Invoice deta entry has six different screen formats. The first screen format prepares the top portion of an invoice. The Accounts Receivable file provides a billing address, while shipping data must be entered manually.

A special file pre-defines up to 10 different types of payment terms which are selected by code when the heading is prepared. This should be adequate for most firms. In addition the same file includes information about the contents and size of the company's packing labels. This portion of the file is used to generate carton labels.

There is no provision to record sales by salespersons or their commissions. No doubt this could be added if required; there is no shortage of memory.

After completing the heading, a transaction entry screen is displayed. This screen allows up to 10 line items. However since the invoice cannot determine the accuracy of the item description and price. With larger disk space such an extension would be possible.

Separate Screens

Setting up a separate screen for each element of the invoice should facilitate the development of custom data entry modules for each business environment. In the test sample each detail data line provided for the following:

A 10 digit compound SKU/part number
A 20 character part description
Unit prices up to 999998.00
Quantities up to 9999.99
An automatic price times quantity extension

You can also include comments such as Partial Order or any other special notation in the body of the invoice.

After all detail lines are entered, an edit screen is presented. The operator may edit the details lines and enter sales tax codes and shipping charges. If the shipping cherges are not available at this time, they can be bypassed and entered later. This option permits you to prepare e preliminary invoice.

The final invoicing screen allows the operator to record or cancel the invoice. Options are also available to selectively alter the heading, detail line items and total—without requiring display of the other portions of the invoice.

The invoicing procedures added by the SBSG to the Osborne system are well thought out. However, because invoicing and merchendise selection are labor intensive activities for most firms, I recommend you customize this application. While this adds to the cost, the resulting labor savings can be significant.

I would like to thank those readers who have been sending "war" stories and letters of encouragement. It is good to hear from you. To those who have been critical about my Model II bias, I hope to review several Model I packages in the near future.■

EDUCATION 80

by Earl R. Savage

A well known quotation states: "There is nothing new under the sun." In spite of that, there are different ways to combine the known in order to accomplish new results. So, let's see what old things we can combine to overcome a couple of frequent problems in instructional programming.

A very common problem is limited memory. My correspondence with instructors around the country indicates that the typical TRS-80 setup has 16K of memory. More limited are the large number of 4K machines in schools and homes. Even a 16K memory can put severe restrictions on an instructional program.

Two Small Programs

The severity of the memory problem is dependent, of course, on the subject matter being taught and the level at which it is being presented. You might want to break up the progrem into two or more smeller ones which will fit the memory. This approech leeves something to be desired even when autometic CLOADing of the sequentiel programs is provided.

The second problem concerns learning styles. We all know that some students learn better by reading, others by listening, others by writing, and so on. It follows that a program designed for general use will be more effective if it provides for more than one learning style. The greater the number of learning styles for which a progrem makes provision, the more effective it will be.

One significant input to the student, overlooked in computer programming, is his auditory sense. Both in school and out, people have been learning for years by means of audio tape recordings. Schools ere well stocked with cassette recorders and instructional tapes. Yet when they get an 80, it seems not to have occurred to them to use the included cassette machine for audio instruction as well as computer programming.

This combination is particularly applicable to programs containing relatively large explanations. If that material requires no interection by the student, there is no point in using valuable RAM memory to contain it. Here's how it all fits together.

The computer program is written in the normal manner, except that long explanations are omitted. After the program is recorded on cassette, an appropriate series of voice recordings is put on the same cassette in the proper sequence. To use the progrem, the learner CLOADs the computer portion of the type. Then, he removes the computer plug from the earphone jack of the cassette machine. The student types RUN and the computer program begins as usual.

When commentary is needed, the computer turns on the cassette and the audio plays out of the speaker. As the program continues, the audio is turned on and off.

Each word of the audio material saves several bytes of RAM which can be used for a treatment of e longer topic in the normal display interaction mode.

The Mechanics

The computer program and the audio may be put on the same cassette or on two different ones. The following instructions are presented as though a single cassette is used.

- In the introduction of the progrem, the students should be instructed to remove the computer plug from the earphone jack and to leave the cassette machine in the play position.
- At each point in the computer program where audio is needed, insert this line:

Digital IC Probe & Logic Pulser

PRB-1 DIGITAL LOGIC PROBE

Compatible with DTL, TTL, CMOS, MOS and Microprocessors using 4 to 15 V power supply. Thresholds automatically programmed. Automatic resetting memory. No adjustment required. Visual indication of logic levels, using LED's to show high, low, bad level or open circuit. Logic and pulses. Highly sophisticated, shirt pocket portable (protective tip cap and removable belt clip).

- Automatic threshold resetting
- Compatible with all logic families 4-15 VDC
- Supply 0.7 V to ± 7 VDC
- No switches/no calibration
- Open circuit detection
- Rugged extended V: 16-28 VDC with optional PA-1 adapter
- DC to > 50 MHz
- 1 nSec pulse response
- 120 K Ω impedance
- Automatic pulse stretching to 50 Msec.
- Automatic resetting memory

PLP-1 LOGIC PULSER

The PLP-1 logic pulser will superimpose a dynamic pulse train (20 pps) or a single pulse onto the circuit node under test. There is no need to unsolder pins or cut printed-circuit traces. Even when these actions are normally performed by digital outputs.

PLP-1 has multi-mode, high current pulse formation packaged in a hand-held, dart pointed portable instrument. It can source or sink sufficient current to force a stalled output condition in digital circuits into the opposite logic state. Signal injection is by means of a push button switch near the probe tip. When the button is depressed, a single high-going or low-going pulse of 2μ sec wide is delivered to the circuit node under test. Pulse polarity is automatic. High nodes are pulsed low and low nodes are pulsed high. Holding the button down delivers a train of pulses (20 pps) to the circuit under test.

- High input impedance (3 state) 1 meg ohm
- Low output impedance (active state) 2 ohms
- Output pulse width 2 μsec nominal
- Input over voltage protection +50 Volts
- Finger tip push button actuated
- Power line reversal protection
- Multi-mode: single pulse or pulse trains
- Automatic polarity sensing
- Automatic current limiting, 7 amps nominal
- Automatically programmed output level
- Circuit powered
- No adjustments required
- Multi-family: DTL, TTL, CMOS, MOS and Microprocessors

PRB-1	DIGITAL LOGIC PROBE	$36.95	PA-1	HIGH VOLTAGE ADAPTER	$5.50
PC-1	POWER CORD, Alligator Clips	$4.95	PT-2	REPLACEMENT PROBE TIP (2)	$1.50
PCB	POWER CORD, Mate-N-Lok	$9.95	PLP-1	LOGIC PULSER	$45.95

OK Machine & Tool Corporation

3455 Conner St., Bronx, N.Y. 10475 U.S.A.

Tel. (212) 994-6600 Telex 126091

*Minimum billing $15 + cost shipping charge $2.00

New York State residents add applicable tax

```
GOSUB6000
```

• Put this subroutine in any appropriate place in the program:

```
6000 OUT 255,4
6010 PRINT@965, "AUDIO ON. . .";
6020 IF INKEY$ < > "G" THEN 6020
6030 PRINT@965, STRING$(15,32);
6040 OUT 255,0
6050 RETURN
```

Each time program execution transfers to the subroutine, line 6000 starts the cassette motor. Line 6010 puts a message at the bottom of the screen and line 6020 stops execution of the program while the audio is playing. When the letter G is typed, execution falls through to line 6030 which removes the audio message from the screen. Line 6040 stops the cassette motor and line 6050 transfers execution back to the main program.

• When you have finished writing and debugging your program, CSAVE it. Then, with the cassette machine completely disconnected from the 80, record the audio segments in the appropriate sequence.

• The first audio segment should be concluded with words similar to these: "It is almost time to return to the other part of the program. When you hear the beep tone, press the letter G on the keyboard. When the beep tone sounds, you should press the letter G for GO. . . . . (BEEP)."

• Each subsequent audio portion should be ended similarly.

• The cassette will continue to run until the letter G is pressed. The space between your audio segments should be sufficient to allow time for the student to find and press the G.

Summary

This new combination of interactive programming and recorded audio segments can be advantageous in many different applications. It works well in almost any type of formal or informal instructional program. The method can be used to list rules and conditions in game programs. Business programs, too, often contain a considerable amount of explanatory material.

The computer/audio technique is great for pointing out the major facts about a chart on the screen, adding sound effects, putting sound and printed words together in the study of phonics, sound and notes in music and so on.

Give the computer/audio program technique a try on your next project. You'll discover just how easy it is to multiply your memory.■

THE ASSEMBLY LINE

by William Barden, Jr.

In the early 60's, I attended an assembly language class which used a Scientific Data Systems computer. One of my fellow students was asked to key in his version of a homework assignment from the control panel of the computer. The instructor then asked him if he was confident that the program would work. The student replied that it would work because the ERROR light on the control panel didn't come on as he entered the machine language program.

These days we are all more sophisticated about program debugging than that. Debugging however, remains just about as tedious and frustrating as it was then. In this column we'll take a look at the general technique of debugging assembly language programs, the debugging tools available and final testing of programs.

Using T-BUG

T-BUG is Radio Shack's cassette-based debug package. It provides rudimentary debugging functions, but can be used effectively to debug programs of any length. T-BUG normally occupies RAM from location 4380H to 497FH. Many people have relocated T-BUG to different memory areas by disassembling T-BUG, observing which instructions were non-relocatable and changing addresses accordingly. This was mostly done early in the TRS-80 game when there was only T-BUG and Small Systems Software RSM-1 available for debugging.

T-BUG can be put onto disk by relocating it to upper memory (above 6FFFH) and using the DUMP command of TRSDOS to write it out as a CIM (core image module). You'll probably want to use the disk DEBUG package instead, and I'll continue to assume so in the following discussion.

Let's assume that you have T-BUG on cassette or disk and want to debug an assembly language program. First, get the object of your program and T-BUG into memory at the same time, by using the following procedure:

1 ORG your program at an area that does not conflict with T-BUG. Assemble it, check for errors, edit and reassemble if necessary and create an object tape on cassette.

2 Load the object tape you created by using the Level II monitor mode. Type SYSTEM after the > prompt of Level II and then type NAME after the *? prompt of the monitor mode to load the object file. NAME is the name you used in assembling your program. If you did not use a name, NONAME is used as a dummy name.

3 Load T-BUG by typing TBUG after the *? prompt that follows a successful load of your object tape.

4 Type / ENTER after a successful load of T-BUG. You should now be in T-BUG as evidenced by a clearing of the left section of the screen and the # prompt.

An alternative to this is to load T-BUG and use it to key in the machine language code for a program to an area of memory. This is useful if you find a listing in *80 Microcomputing* or elsewhere and don't want the agony of entering and assembling the source code. Make certain that the location of the program doesn't conflict with the T-BUG area, and, that all the code is there. T-BUG shines at rapid entry of machine language bytes; you can enter them as fast as you can type!

First Steps in Debugging

You've got your program and T-BUG in memory and are in T-BUG. Much of your debugging should have been done already! You should have gone through your listing several times in detail and "desk-checked". Assembly language programming is not interactive; if you find errors, you'll have to edit, reassemble, and reload, and you'd like to keep that to a minimum.

Table 1 lists the T-BUG commands available. Basically, all you can do is examine memory locations and register contents, set breakpoints, and read and write cassette tape files.

Is it possible to debug effectively with such a limited number of commands? From my experience with T-BUG, DEBUG (disk), RSM (Small Systems Software), Z-BUG (MicroSoft's EDTASM-PLUS Debug), and a number of minicomputer and large computer debug packages (half of which seem to be named so that their initials spell out "DDT"), I would have to say yes.

I would say that the time spent debugging a 1000-line program with T-BUG vs. the time spent with the most powerful de-

#B aaaa	Set breakpoint at hex location aaaa
#F	Fix* previous breakpoint. Use after breakpoint.
#G	Continue from breakpoint.
#J aaaa	Jump to hex location aaaa.
#L	Load a T-BUG or SYSTEM tape.
#M aaaa	Display location aaaa. Enter new value if contents to be changed or simply ENTER if OK.
#P aaaa bbbb cccc NAME	Write cassette from aaaa through bbbb with starting address cccc and file name NAME.
#R	Display registers
X (after M, J, B, P)	Exit operation.

Table 1. T-BUG Commands

bugger would be no more than twice as long. One does not continually enter a stream of commands to the debug package—there's a lot of head scratching going on in between. The exception to this might be MicroSoft's Z-BUG, where editing and reassembling can be done without reloading on an interactive basis, enabling efficient program development.

The procedure commonly used with any debug package is this: First, every programmer tries one run to see if by some miracle it works just as expected the first time. (It doesn't.)

Next, a search for gross errors is done. This is not a systematic procedure, since there will probably be bugs popping out at you on execution. Use the B command to set a (B)reakpoint and then execute a J(ump) to the start of your program. The breakpoint is exactly that—the program will be executed until the breakpointed instruction is reached and then T-BUG will be re-entered. This gives the user control so the program doesn't bomb. If a program hang-up occurs, the program and T-BUG will have to be reloaded, or it may be possible to RESET the CPU and restart T-BUG at location 43A0H (by SYSTEM and 17312/).

T-BUG implements the breakpoint by putting a CALL 4380H into the breakpoint location. This can have disastrous results (Fig. 1), where the 43H wipes out a variable used earlier!

When the breakpoint is reached, variables, buffers, or other memory locations can be examined for proper contents by using the M(emory) command to display memory locations. The R(egister) command displays register contents. The M command can modify any location by typing in a new value in hex.

One of the failings of T-BUG is that registers must be modified by altering memory locations associated with them.

The F(ix) command restores the original values to the breakpointed location. After the breakpoint is fixed, a G(o) can be used to continue from the breakpoint after a new breakpoint is established.

Binary Search for the Next Error

Debugging using T-BUG proceeds in this fashion: establishing one breakpoint at a time, reaching it, examining variables and buffers for proper results, and establishing a new breakpoint. The process evolves into a binary search for the next error—breakpoint halfway through, see if the breakpoint is reached. If not, establish one earlier, and so forth. This is not sophisticated debugging, but it works.

The P(unch) and L(oad) commands can be used to write and read in T-BUG cassette files. T-BUG files have a format identical to SYSTEM tapes produced by EDTASM. If you're working with a large assembly language program, it's convenient to patch and save the program on cassette every so often. This way the patched program can be reloaded. Since T-BUG can be saved in the same tape file, a P(unch) command can produce one enormous file including the patched program and T-BUG; this can then be reloaded with a single SYSTEM command.

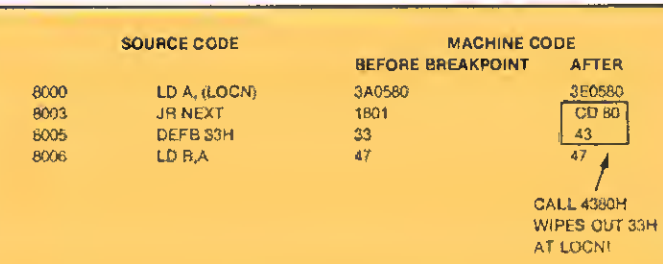


Figure 1. T-BUG Breakpoint Problem

Patching

Patching is the process of deleting, modifying, or inserting machine code directly to the object or machine code in memory without reassembling. Here's an example: Suppose we want to add two instructions after PATCH in the program of Fig. 2. Obviously there's no room between the instructions (or little, anyway). The instructions are added to a patch area somewhere in memory and the code is modified as shown.

Here's a philosophical question—when should you patch and when should you reassemble? You should certainly patch if you are sharing a TRS-80 with 32 other programmers and you can't get on the machine to reassemble for six days. You should certainly not patch if you are using EDTASM-PLUS with in-memory assembly capability.

For all other conditions, you should patch whenever you find errors that can be corrected by modifying one instruction (such as changing the register in LD R1,R2), by deleting one or more instructions, or by inserting instructions. Reassemble whenever you have patches of more than a dozen or so.

To patch you must do some hand assembly of instructions. Another way to find the proper instruction configurations without manual assembly is to look through your listing to find identical or similar instructions for the patch. The patch area may be adjacent to the program, or it may be anywhere in RAM. If it is close to the program area, it is easier to include it in a P(unch) command.

Using DEBUG

If you have a disk system the debug task is more convenient. DEBUG can be called off disk. It loads into the system utility area. If you are using the Apparat EDTASM, MISOSYS EDTASM, or the Radio Shack Disk Assembler, source files can be saved on disk and object files written to disk. The latter feature makes it easy to reload the machine code for debugging purposes. The sequence for loading the object and transferring control to DEBUG goes like this:

1 After TRSDOS DOS READY, prepare DEBUG by typing DEBUG. TRSDOS will come back with DOS READY again.

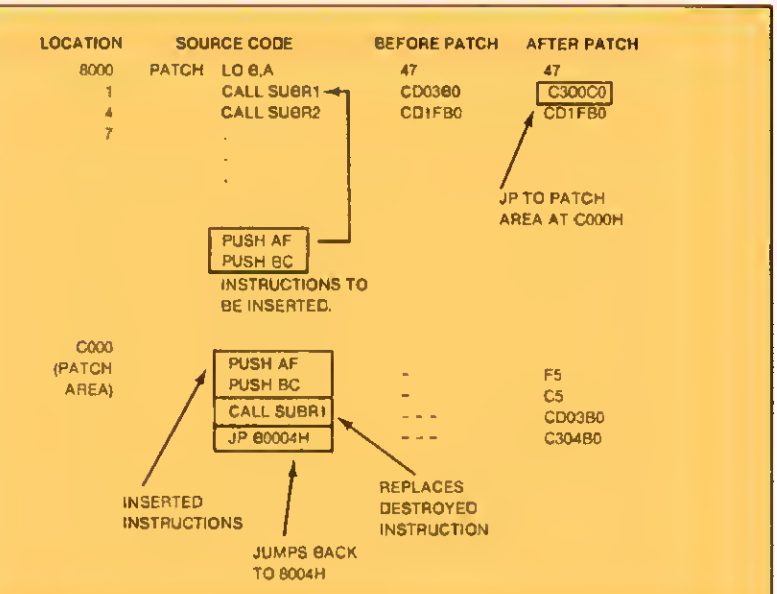


Figure 2. Patching Code

A	Display in ASCII.
C	Single step instructions, but execute CALLs in full.
Daaaa (SPACE)	Display from hex location aaaa.
Gaaaa (,bbbb(,cccc))	Execute at location aaaa with optional breakpoints bbbb, cccc
H	Display in hexadecimal.
I	Single step instructions.
Maaaa (SPACE)	Set modification address to aaaa and display data.
Rrp dddd (SPACE)	Load register pair with dddd.
S	Full screen display
U	Set dynamic display update mode
X	Cancel command, set display to register mode
;	Increment memory display to next block.
–	Decrement memory display to last block.

Table 2. DEBUG Commands

2 Load the object module output by Apperat, MISOSYS, or the RS disk assembler by LOAD NAME.

3 Hit BREAK to enter DEBUG.

The DEBUG commands are shown in Table 2. DEBUG permits the same operations as T-BUG, but displays a screen full of memory contents or a combination of memory and register contents. More than one breakpoint may be specified, which is a decided advantage as there is invariably more than one path through the program to be checked.

After each patch has been made, the patched core image can be written out to disk by performing a G402D, which reboots TRSDOS, and then doing a DUMP NAME (START = X'aaaa',END = X'bbbb', TRA = X'cccc'). Be sure to include the patch area in the area to be DUMPed. The DUMP will create a new NAME file or replace the old NAME file which can then be LOADed as before.

As with T-BUG, too many patches get confusing, and at some point it's best to do a new edit and reassembly.

DEBUG includes a command to single-step instructions one at a time. Single-step through Level II or TRSDOS routines to supplement information gleaned from disassemblies. Single-step can be also used to trace a path through your program to find out where a variable gets clobbered, or when an unexpected path is taken.

Small Systems Software RSM-2

Small Systems Software was one of the first companies to bring out a significant piece of software for the TRS-80. Their RSM-2 and RSM-2D are upgraded versions

Give Your Computer the Best

The Microline 80

You can't find a better small printer. The Microline 80 will outperform and outlast every competitor. It will run all day at 80 cps with no duty cycle limitations, producing letter perfect printing on plain paper. And the head is warranted for 200,000,000 characters.

You can't find a better value either. The Microline 80 includes upper and lower case characters, double width and condensed printing, friction and pin feed, six and eight line per inch spacing and block graphics for charts, graphs and diagrams. And it operates with TRS-80™, Apple™ and other popular small computers. The only extras are snap-on tractors and a buffered RS232 interface. Give your computer the best, the Microline 80.

OKIDATA

Okidata Corporation
111 Gaither Drive
Mount Laurel, New Jersey 08054
609-235-2600

of the original RSM-1 monitor, and they are powerful monitors. Monitor here is synonymous with debug package rather than the earlier meaning of control program or operating system.

The RSM-2 and RSM-2D packages contain commands to display and modify memory, to transfer control, and to breakpoint as in T-BUG and DEBUG. The packages also contain many other useful commands, such as FIND (search block for one byte), HUNT (search block for address value), MOVE (move one block to another), TEST MEMORY (random memory test), EXCHANGE (exchange two blocks of memory), ZERO (fill memory block with specified byte) and others.

The packages allow display of memory in ASCII or hex, printing of data to a parallel line printer (or to a serial printer through a serial interface), reading and writing SYSTEM tapes, reading and writing disk sectors (RSM-2D), and, for a grand finale, include a Z-80 disassembler for displaying code in mnemonics. When disassembling, none of the original comments from any code are printed!

A Zbignew Z-BUG Package

I don't want to keep harping about certain products—but after all, even I have a price (roughly equivalent to a box of diskettes)... The Microsoft EDTASM-PLUS, however, really is a superlative package. (Alas, it is cassette and not disk based). It includes a beefed-up Editor, Macro Assembler, and most importantly, the ability to assemble directly in memory. The last feature allows the debug portion of EDTASM-PLUS to be used on an interactive basis with the Editor and Assembler. The object code of a program can be debugged, and an immediate edit and reassembly can be done without reloading.

Z-BUG includes most of the features mentioned above, including disassembly and single stepping. Its single, most powerful feature operates in conjunction with in-memory assembly—symbolic debugging.

When an assembly is performed, the machine code is automatically assembled in the next available (or user specified) section of memory. At the same time, the assembler symbol table is preserved. This symbol table can be referenced by Z-BUG to examine memory locations symbolically. For example, you can type "TABLE/", and Z-BUG will search the symbol table for the location of TABLE, and then display its contents. Data can also be input in symbolic form—a location could be modified to the value LOOP + 5, for example.

Program Final Testing

The last step of the debugging process should be a comprehensive test of the final version of the program. A basic programming maxim is that programs never work the first time. Here's another: There is no final program!

Programs are released with bugs for two reasons. The first is in the nature of programs themselves. Programs are designed to provide generic solutions to many permutations of inputs and outputs. Not all permutations can be tested—there are simply too many possibilities. As a result, programmers pick representative inputs and outputs for testing. In the worst cases, a few runs are made through the program and the program is then pronounced "tested". In the best cases, a test plan is drawn up and the program is tested by a test driver. It's entirely possible that the final testing phase could take 25 percent of the total time spent developing the program! I'd like to recommend the programs of a company that does this comprehensive final testing, but they've unfortunately gone out of business...

Which leads us to the second reason there are bugs in final versions of TRSDOS, NEWDOS, VTOS, Level II BASIC, and just about every other program. As every programmer working in a commercial environment knows, there is always a great deal of pressure to finish a program so that it can be sold. This holds true in TRS-80 software companies as well.

We'll just have to live with the bugs, ferret them out, and hope that the companies correct them. Meanwhile, make it a goal to do some final testing of your assembly language programs. End of sermon.

Still Another Model I Assembler

Roy Soltoff of MISOSYS sent me a copy of his MISOSYS Disk Mod. (I suspect he wanted me to use it, like it, and write about it in this column.)

The Disk Mod is a set of patches for RS EDTASM that converts it to a disk assembler with source and object file storage on disk. Other features I found handy were the ability to interface a serial printer, and page formatting. In short, this version of EDTASM contains all of the Apparat changes to EDTASM plus others. I've used this and I like it. (OK, Roy, you can send that box of diskettes).

Next month we'll have the results of the Third Assembly Line Programming Contest. (I'm getting the Amana ready for shipment to the winner now... ■

80 INPUTS

Continued from page 28

Printer Pagination

One of the less than desirable features of Radio Shack's Printer I with the roll paper is that there is no way to get page spacing when LLISTing a long program or printing a long calculation report. At least, there is nothing in the documentation to cover this.

Dr. Lien's "Learning BASIC II" tells us that the standard printer page length is 66 lines and that this quantity is stored at memory location 16424. Also, stored at location 16425 is the variable that tells the computer how many lines the printing head has moved away from the last top of form positioning. The command "LPRINT CHR$(12)" moves the printing head to the next top of form and restores memory location 16425 to 0 to start recording the new page.

Try this little routine. Disk save a long program (more than 100 lines) in ASCII—save "BUDGER/BAS", A. Now it can be read and inputted as a sequential file. Now run this little program:

```
10 CLEAR 500
20 OPEN"I",1, "BUDGER/BAS"
30 FOR N = 1 TO N:'N = YOUR NUMBER OF LINES
40 IF PEEK(16425) = 50 THEN LPRINTCHR$(12) ELSE 50
50 LINEINPUT# 1, R$
60 LPRINT R$
70 NEXT N
80 CLOSE
90 LPRINT CHR$(12)
```

There it is, your long printout is paginated and you can fold it or rip it into equal pages, side punch, and store in a binder. Many other things can be done with PEEK (16425)!

Richard Halloran
San Francisco, CA

Qwikdisk

In your article in the September issue called QWIKDISK, the 09H and 19H numbers gives 12ms step times not 10ms. The 08H and 18H numbers also give only 12ms not 5ms step times.

This information is based on Western Digital's data sheets for FD1771-01 Floppy Disk Formatter/Controller and my own experiments.

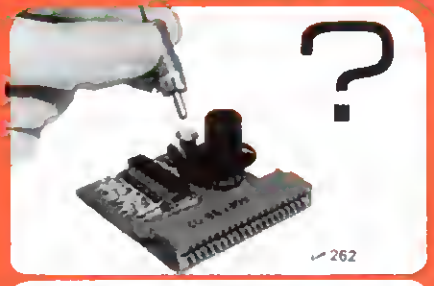
Eric Espenhahn
Lake Park, FL

OUR CHRISTMAS GIFT:

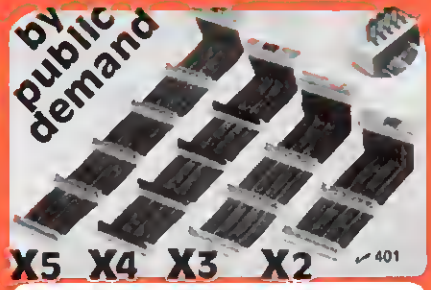
THE ALPHA GREEN SCREEN AT HALF-PRICE

WITH ANY PURCHASE

STICK-80 MAKES KEYBOARD OBSOLETE.
Features the famous ATARI Joystick 8 directions + fire control. Simple instructions to make joystick versions of most action games. Plugs into keyboard or expansion int. Price includes ATARI joystick with ALPHA interface and instructions. FREE "MAGIC ARTIST" program. $29.95
Super Real Time Action Graphic Sound games for Stick-80 by Software Innovations: ALIEN INVASION, CAR RACER, COSMIC INTRUDERS, BREAKOUT. Each $19.95
STELLAR ADVENTURE: super action with sound $14.95
Software authors and distributors: contact us for joystick conversion package for your existing games.



MUSIC-80 MUSIC-80 MUSIC-80 MUSIC-80 MUSIC-80
Use existing software
or write your own. With this low cost 8 bit digital to analog converter you can synthesize up to 5 music voices. Built-in volume control handy when stereo not near TRS-80. Simply plug the "MUSIC-80" into the keyboard or the E/I screen printer port and connect the output (RCA jack) to any amplifier. The Radio Shack $12 speaker/amplifier works fine.
Fully assembled and tested, 90 day warranty. $39.95

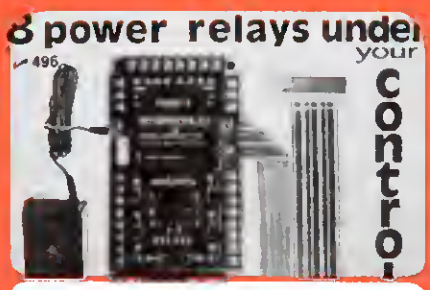


YOU ASKED FOR IT: "EXPANDABUS" X1, X2, X3 AND X4
CONNECT ALL YOUR TRS-80 DEVICES SIMULTANEOUSLY on the 40 pin TRS-80 bus. Any device that normally plugs into the keyboard edge connector will also plug into the "EXPANDABUS". The "X4" is shown with protective covers (included). The TRS-80 keyboard contains the bus drivers (74LS367) for up to 20 devices, more than you will ever need. Using the E/I, it plugs either between KB and E/I or in the Screen Printer port. Professional quality, gold plated contacts. Computer grade 40 conductor ribbon cable.
X2 $29 X3 $44 X4 $59 X5 $74
Custom configurations are also available: call us.

ANALOG-80: A WORLD OF NEW APPLICATIONS POSSIBLE
8 DIGITAL MULTIMETERS PLUGGED INTO YOUR TRS-80!!! Measure Temperature, Voltage, Current, Light, Pressure etc. Very easy to use: for example, let's read input channel #4: 10 OUT 0,4: 'Selects input #4 and also starts the conversion. 20 A = INP(0): 'Puts the result in variable "A". Voila! Specifications: Input range: 0-5V to 0-500V. Each channel can be set to a different scale.
Resolution: 20mV (on 5V range). Accuracy: 8 bits (.5%). Port Address: jumper selectable. Plugs into keyboard bus or E/I (screen printer port). Assembled and tested, 90 day warranty. Complete with power supply, connector, manual. $139

INTERFACER 2: LOW COST INPUT/OUTPUT MODULE
Still the best value in sense/control devices. Use it for energy control, burglar alarm, darkroom, selectric drive, model trains, robots, Skinner box.
— 8 latched TTL outputs, 2 relays SPDT 2A, 125V contacts
— 8 TTL/CMOS inputs. Input 0 and 1 are optically isolated
— Neat and compact design, very easy to use
10 A = INP(0): 'Reads the 8 inputs (if A = 0, all inputs are low). 20 OUT 0, X: 'Controls the outputs and the relays.
Assembled & tested, 90 day warranty. Price includes power supply, cable to KB or E/I, superb user's manual, free phone dialer program. $95. Manual only $5.

LET THE "CHAIN BREAKER" FREE YOUR MINI-DRIVES
End the daisy-chain mess once and for all. Fits all mini-drives: Porcom, Aerocomp, Shugart, Micropolis, MTI, Vista, Pertec, Siemens, BASF. Easy to install: just remove the drive cover, plug in the "CHAIN BREAKER" and replace the cover. Voila!!!
Now you can change and move your drives around without disassembly. Keep the cover on and keep the dust out. High reliability, gold plated contacts, computer grade 34 conductor cable. Tested and guaranteed.
Get one for each drive ... only $13.95



INTERFACER-80: the most powerful Sense/Control module
- 8 industrial grade relays: single pole, double throw, isolated contacts. 2 Amp @ 125 Volts. TTL latched outputs are also accessible to drive external solid state relays.
- 8 convenient LEDs constantly display the relay states. Simple "OUT" commands (in basic) control the 8 relays.
- 8 optically-isolated inputs for easy direct interfacing to external switches, photocells, keypads, sensors, etc. Simple "INP" commands read the status of the 8 inputs. Selectable port address. Clean, compact enclosed design. Assembled, tested, 90 days warranty. Price includes power supply, cable, connector, superb user's manual. $159

WARNING

IBM and all the "biggies" are using green screen monitors. Its advantages are now widely advertised. We feel that every TRS-80 user should enjoy the benefits it provides. But **WARNING**: all Green Screens are not created equal. Here is what we found:

- Several are just a flat piece of standard colored Lucite. The green tint was not made for this purpose and is judged by many to be too dark. Increasing the brightness control will result in a fuzzy display.
- Some are simply a piece of thin plastic film taped onto a cardboard frame. The color is satisfactory but the wobbly film gives it a poor appearance.
- One "optical filter" is in fact plain acrylic sheeting.
- False claim: A few pretend to "reduce glare". In fact, their flat and shiny surfaces (both film and Lucite type) ADD their own reflections to the screen.
- A few laughs: One ad claims to "reduce screen contrast". Sorry gentleman but it's just the opposite. One of the Green Screen's major benefits is to increase the contrast between the text and the background.
- Drawbacks: Most are using adhesive strips to fasten their screen to the monitor. This method makes it awkward to remove for necessary periodical cleaning. All (except ours) are flat. Light pens will not work reliably because of the big gap between the screen and the tube.

Many companies have been manufacturing video filters for years. We are not the first (some think they are), but we have done our homework and we think we manufacture the best Green Screen. Here is why:

- It fits right onto the picture tube like a skin because it is the only **CURVED** screen **MOLDED** exactly to the picture tube curvature. It is Cut precisely to cover the exposed area of the picture tube. The fit is such that the static electricity is sufficient to keep it in place! We also include some invisible reusable tape for a more secure fastening.
- The filter material that we use is just right, not too dark nor too light. The result is a really eye pleasing display.

We are so sure that you will never take your Green screen off that we offer an unconditional money back guaranty: try our Green Screen for 14 days. If for any reason you are not delighted with it, return it for a prompt refund.

A last word: We think that companies, like ours, who are selling mainly by mail should list their street address, have a phone number (for questions and orders), accept CODs (not every one likes to send checks to a PO box), offer the convenience of charging their purchase to major credit cards. How come we are the only green screen people doing it?

Order your **ALPHA GREEN SCREEN** today. $12.50
Or enjoy our Christmas gift and pay only $6.25 when you order anything else.

497

ALPHA Product Co.

ADD $2.50 PER ORDER FOR SHIPPING AND HANDLING
ALL ORDERS SHIPPED FIRST CLASS MAIL
WE ACCEPT VISA, MASTER CHARGE, CHECKS, M.O.
COD: ADD $2.00 EXTRA
QUANTITY DISCOUNTS AVAILABLE
N.Y. RESIDENTS ADD SALES TAX

85-71 79 St. WOODHAVEN, N.Y. 11421 — Info and order: (212) 296-5894

1980 Christmas Sale

Model I & II Software

Where Your TRS-80 Means Business —
The Best in Model I or II Programs Available Now!

MODEL 1
Complete Business System

This integrated system includes Invoicing, Inventory Control, Accounts Receivable, Accounts Payable, Payroll, General Ledger and Word Processing. A complete Business System for your Model 1.

SPECIAL OFFER

$299.95

MODEL II
— Business System —

Complete with all manuals. This is one of the best and most complete Business Systems for your Model II. Best of all it is easy to use. Enter the world of the Model II with a system that works!

SPECIAL OFFER

$649.95

MODEL I
Medical/Dental System

Complete System — Easy to use! Now you can use your TRS-80 where you intended to use it — in your office! Give us a call for complete details on this new Complete Package.

SPECIAL OFFER

Special **$349.95**

MODEL II
CP/M Release 2.2

Including Utilities and full documentation. This is the CP/M designed for the Model II.

SPECIAL OFFER $169.95

Including CBASIC-2 **$229.95**

CBASIC-2 ONLY **$79.95**

Before You Buy Any Software Call Us First!

★ SPECIAL ★
Verbatim 5¼" Mini Disks

$24.50 Box (10 Per Box)

Verbatim 8" Floppy Disks

Double Density FD34-8000)

$44.50 Box (10 Per Box)

Limit 4 Boxes Per Customer

*OFFER GOOD WHILE QUANTITIES LAST

MODEL II
Medical/Dental System

Same as above with much more! This System also includes our computer-Based Patient History System. THE BEST!

Special **$699.95**

★ Christmas Gift

5 Free Data Diskettes with each Complete Business System

★ All Programs Supplied on Diskette ★

New

Hotel/Motel System

$799.95

Property Management System

$549.95

MODEL II
MAGIC WAND
— Word Processing System —

This System offers the best features of any system on the market, including the best documentation.

Special $289.95

STRUCTURED SYSTEMS GROUP

General Ledger	$729.95
Accounts Receivable	$729.95
Accounts Payable	$729.95
Payroll	$729.95
Inventory Control	$429.95
Analyst	$189.95
Letterright	$165.95
NAD	$ 79.95
QSORT	$ 79.95

MICROPRO

Word Star (Ver 2.1)	$329.95
Word Star w/Mail-Merge	$479.95
Data Star	$279.95
Word Master	$119.95
SuperSort I	$189.95
SuperSort II	$159.95
SuperSort III	$119.95

MANUALS

OSBORNE/McGRAW HILL

Accounts Receivable/ Accounts Payable	$18.95
Payroll w/Cost Accounting	$18.95
General Ledger	$18.95
CBASIC 2 Manual	$12.95
Structured Systems Manual	$24.95
Graham Dorian Manual	$34.95
Magic Wand Manual	$34.95
Word Star Manual	$34.95
CP/M Handbook (Sybex)	$12.95

GRAHAM-DORIAN

General Ledger	$779.95
Accounts Receivable	$779.95
Accounts Payable	$779.95
Payroll	$479.95
Inventory Control	$479.95
Invoicing/Order Entry	$779.95
Cash Register	$479.95
Apartment Management	$479.95
Job Costing	$779.95

Looking for a Specific Program at the Best Price — Call Us Today!
Thinking Business - Take Advantage of these Limited Offers.

We now handle software to support many microcomputers other then the TRS-80*

Software-**Mart** 286

24092 PANDORA STREET
EL TORO, CALIFORNIA 92630

VISA

24 HOUR HOT LINE

(714) 768-7818 (in California)
1-800-854-7115

SOURCE MAILBOX: TCU155

Give us the Opportunity to Beat any Nationally Advertised Price!

"OUR BEST AD'S ARE NOT WRITTEN — THEY'RE RUNNING ON TRS-80's"

*Quantities limited on some items ● TRS-80 is a trademark of Radio Shack ● Magic Wand is a trademark of Small Business Applications ● CP/M is a trademark of Digital Research, Inc ● All Software is sold on an "as is" basis and without warranty ● Prices and programs are subject to change without notice ● Add $2.00 shipping & handling on each order

OFFER EXPIRES 1/31/81

Reader Service—see page 274

80 REVIEWS

Edited by Pamela Petrakos

"A group of composers, with their ears tuned to the future, had developed musical techniques on early 'monster' computers, and an unfamiliar, disquieting kind of 'computer music' was born."

"1001 Things To Do With Your Personal Computer"
Mark Sawusch
TAB Books, Inc.
Blue Ridge, PA
$12.95 Hardcover, 335 pp.
$7.95 Softcover

by Fred Blechman

Probably the most annoying question aimed at computerists by non-computerists is, "OK, so what can you do with it?" Mark Sawusch addressed this question and came up with over 1000 answers!

Sawusch's 335 page book is an amazing collection of practical ideas and programs divided into 12 broad categories, and includes a glossary, appendix and index. Each category contains at least several, and as many as dozens of potential applications. A run through the chapter titles indicates the enormous scope of this book: Applications for Everyone, Business and Financial Applications, Technical and Scientific Applications, Educational Applications, Hobby Applications, Games and Recreational Applications, Control and Peripheral Applications, Artificial Intelligence and The Future Personal Computer, Utility Programs, Miscellaneous Applications and A Compendium of Additional Applications.

The four page glossary explains the meaning of common computer and programming terminology. The appendix covers financial formulas, gives addresses of 78 microcomputer manufacturers, contains a table of metric conversions and presents 11 flowcharting symbols.

All in all, this book is really overwhelming! Although it contains 75 actual programs (several running over four pages!), this book is not intended as a how-to book as much as a what-to-do book. More than 1000 ideas are offered, and covered in sufficient detail to provide a basis for a virtually unlimited number of spin-off ideas.

I was particularly impressed with the supporting information provided in many of the chapters. Simple formulas are used throughout, so you can easily develop your own programs by building on or altering the example programs provided. Diagrams, flowcharts and illustrations are sprinkled throughout. The type is large and easy to read and the program listings are in bold typeset—*not* hard to read reduced photocopies of matrix printing!

Although I couldn't find mention of the programming language anywhere in the text, it appears that all programs are Radio Shack TRS-80 Level II BASIC, with some programs designed for disk use. This means they can be adapted to the majority of microcomputers that use Microsoft BASIC. There are no machine language or assembly language programs included.

My criticisms of this book are in the programs and listings. Some programs are very long, yet no indication of memory requirements are given. It appears that some would exceed 16K, and sometimes a program that looks short uses extensive string or array space. I would like to see each program with a REM line indicating memory needed, and whether the program can be used without disk.

Also, because each of the programs has been typeset rather than photocopied from an actual listing, there are numerous typesetting errors. This, together with the fact that the author has made no attempt to explain the line by line operation of the programs, and has not listed the variables and their usage, makes this a book too advanced for beginners. While a beginner could certainly key in and RUN the programs, the main thrust of the book is to stimulate ideas for those already familiar with BASIC programming.

My hat is off to Mark Sawusch for the effort and imagination he used writing this extremely stimulating book. If you are into BASIC programming, you'll probably find enough ideas and examples here to keep you and your computer busy for 01100100 binary years! ■

An Introduction to Computer Music
Wayne Bateman
John Wiley & Sons, Inc.
New York, NY
Hardcover, 314 pp.
$24.95

by Dennis Bathory Kitsz

Somewhere, a mechanical voice sings "Daisy, Daisy, give me your answer true." Some time ago, an enthusiastic high school science teacher played us an experimental recording of that song, and, with half chuckle, half sigh, acclaimed it a portent of things to come.

A group of composers, with their ears tuned to the future, had developed musical techniques on early "monster" computers, and an unfamiliar, disquieting kind of "computer music" was born.

But those were the days when robots were imminent; George Orwell's *1984* was hardly a decade old. The experiments in computer music conducted by Lejaren Hiller and others were viewed with a hostility interbred with fear.

Renewed Interest

The appearance of *An Introduction to Computer Music* by Wayne Bateman heralds a renewed interest in the genre.

The real 1984 is now in sight. An Orwellian cataclysm seems nearly as quaint as the predictive fictions of Jules Verne and H. G. Wells.

Maxwell House coffee jingles have brought this musical electronic sound to the public. Robert Moog's "music synthesizer" has made his name as familiar as Kleenex. No amateur band was complete without one; computer music composers were forced to retreat to the safety of the universities.

The production of music generation peripherals for the TRS-80 and other person-

al computers demonstrates that many composers other than academics are now enthused about sound and music created with the aid of digital circuits.

Computer music is not merely electronic music, though, but rather a very versatile technique of composing and orchestrating sound and structure beyond that normally available to humans.

In general, electronic music is any sort of deliberate sound created or mutated by electronic means, issuing from a loudspeaker. In classical terms, electronic music can be divided into three overlapping areas:

- **Concrete music:** The original music is acoustic, meaning it is produced without electronic help. Then, that sound is transformed by electronic circuitry.
- **Synthesized music:** Many musicians object to this term, claiming that all music is real, not synthetic (in fact, I call my own synthesizer an "electronic music developer" to get away from that artificial music term). But the phrase can be generally defined as any music originated by electronic means and processed through traditional audio circuitry (oscillators, filters, reverberators).
- **Computer music:** This music is generated, manipulated and controlled by a computer. Normally, only the final presentation to the listener involves any analog (audio) circuitry.

Bateman's book deals exclusively with computer music. The book is neither academically thorough nor popular, occupying a dangerous middle ground in which Bateman is not entirely comfortable. Bateman is a lucid writer, but the topic is too big. *Introduction* leaves us confounded by detail.

The question of the computer's validity as a musical device is briefly discussed in the first chapter. Bateman believes in that validity, and presents the physical and mathematical fundamentals of its tones and their harmonics. Frequency spectra, additive and subtractive synthesis of complex tones, sampling intervals and phase relationships are presented. These topics are complex, but vital to computer composition, so Bateman includes a formidable but inevitable helping of mathematics.

Two unsatisfying chapters on computer operation and languages follow. (Bateman's machine has the unnerving habit of giving its accumulator a compliment, rather than complementing it.) These chapters present flow charts and theoretical programs in FORTRAN, Pascal and "English." BASIC program samples are in the appendix. The author does not tell what machinery to use to test his theoretical programs, on the assumption that the hardware (but not the software!) might become outmoded. This leaves the reader unclear on how to "plug in" to the computer.

Waveform Analysis

Successive topics include modulation (not musical, but sonic), dynamics and waveform analysis. The chapters contain a great number of graphs representing sonic events. The waveform analysis chapter is Bateman at his best, but even the experienced composer/programmer winces at the convoluted waveforms of oboe and clarinet, for which separate charts are presented for each of the first twenty-one partials!

Bateman describes the computer's synthesis of complex tones—sounds which cannot be created by sounding objects, but are the results of waveforms, manipulated and reformed, within the composer's mind.

He asserts that this changes the way a composer will create new works of art. "Here, the composer is in direct control of the timbral quality of all the sounds in the composition. Consequently, he or she must understand the fundamental constitution of these sounds and the principles governing the methods of their production. This is why extensive study of acoustics and waveform analysis must now take a prominent place in music theory as the electronic medium is brought into the art."

The text discusses recorded and natural sounds, proposing a difference in approach between the more common analog processing and the difficult but potentially more accurate and reproducible method of sound generation with a computer.

Finally comes the art: Scales and tonality are presented with a lucidity and depth of understanding surprising and gratifying. Obviously, Bateman is at home with contemporary Western music and its long history, and his tone and selected musical examples are both to the point and refreshing.

Bateman has included a probing discussion of the dilemmas of the computer in modern society, "Machines and Human Creativity." Bateman speaks of the personification of machines this way: "Anyone who programs a computer quickly becomes accustomed to its cold, mechanistic responses to every instruction, and to its banal incapability of humanistic interaction." Bravo for Bateman.

Introduction to Computer Music remains an unsatisfying work. It is because personal computer users are given no hint on how to begin the task of composition. Because it seems mathematically detailed, the book can be overwhelming. Also it assumes some knowledge of music theory, and is not directed to the growth of the extemporizing composer/performer. But, the book does present a topic returned from public banishment, and deserves the attention of composers and other musicians, as well as computer hobbyists. ■

KEEPIT Version 2.0
Dennis Bathory Kitsz
The Alternate Source (TAS)
Lansing, MI
$9.95

by Jack Decker

Many folks have purchased the TRS-80 Model I in expectation of using it for serious applications only to discover the limitations of the cassette-based system. For those unable to justify the added expense of moving up to disk operation, there is now available a very underrated program that could make serious applications on the Model I a whole lot easier.

Written by Dennis Kitsz (a name that should be familiar to *80 Microcomputing* readers), KEEPIT is a utility program that packs a lot of power into less than 1K of machine code.

Several Features

KEEPIT has several features. First is the inclusion of the KBEEPFIX routine which initially appeared in *80 Microcomputing* (February, 1980, pages 14 and 15, also see the update which appeared in the column on pages eight and nine in April, 1980). This routine provides keyboard debounce, automatic character repeat (after a short delay) when a key is held depressed, and an audible beep at the cassette output port each time a keystroke is entered.

Useful as that may be, the next feature

is the real workhorse of the program because it allows BASIC programs to be saved in the middle of a RUN with all its variables intact.

Here's how it works: At whatever point you want to save your program, you press BREAK. Set up a cassette to record the program, then type the command:

```
*SAVE/RUN"PROGRM"
```

(The asterisk is the cue for the KEEPIT program to take over from BASIC and interpret the following commands.) "PROGRM" may be replaced by any file name of up to six characters.

When you want to retrieve your program, hit ENTER for the MEMORY SIZE?, and ready the cassette recorder. Enter SYSTEM, then enter the file name. The tape will load, and the program will reappear exactly the way you saved it.

What happens is: The BASIC program is re-loaded, along with all variables, systems pointers, the KEEPIT program itself; even the video display is restored just as it was. Only "free space" (memory not used during execution of the program) will not be affected. You simply CONTinue your BASIC program right where you left off!

Such a feature can be used to debug new programs by saving a program at various points throughout a run, thus allowing you to go back and reconstruct what was happening in the logic flow of the program just before the crash. "Epic" game players will find it handy to be able to save a game in progress and return to it at their convenience. (KEEPIT does louse up the current screen display a bit, however this may not be applicable to all games.)

I think its most practical application will be to maintain data in array variables within the BASIC program, and output it to tape along with the program.

As an example of the latter, consider a short program to save ten names and phone numbers to be displayed later on the video. A simple save program might look like this:

```
10 FOR X = 1 TO 10
20 INPUT N$,P$
30 PRINT #-1,N$,P$
40 NEXT
```

Later on, when you want to read the data, you can use this program:

```
10 CLS: FOR X = 1 TO 10
20 INPUT #-1,N$,P$
30 PRINT N$,P$
40 NEXT
```

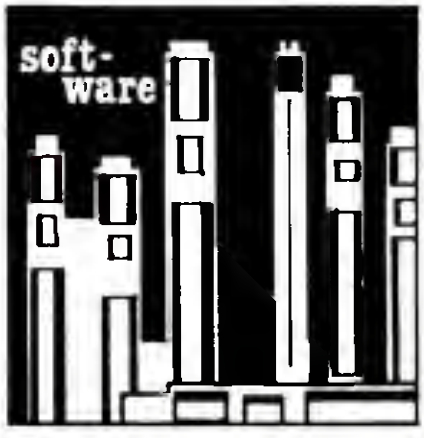
Or, using KEEPIT you can do this:

```
10 FOR X = 1 TO 10
20 INPUT N$(X),P$(X)
30 NEXT
40 CLS: STOP
50 CLS: FOR X = 1 TO 10
60 PRINT N$(X),P$(X)
70 NEXT
```

When the program stops at line 40 (after all entries have been made), you could type:

```
*SAVE/RUN"PHONE"
```

Later you can load "PHONE" as a SYSTEM tape and CONTinue. Notice that this saves you the trouble of loading the program and the data in separate segments, and it saves you the time required to execute several PRINT #-1 and INPUT #-1 statements. When large amounts of data are involved, this can be a real time-saver.



Disk*Mod
MISOSYS
Alexandria, VA
$20

by **Buzz Gorsky**

When I recently acquired my disk system, I began looking for a utility that would make my Radio Shack Editor/Assembler more useful. I had lots of assembly language programs on tape, many of which required editing to make the machine code compatible with the disk system. The thought of having to enter them again into a disk-based system such as Radio Shack's $99 disk EDTASM package was not appealing. When I saw an ad for the MISOSYS Disk*Mod program, I decided to give it a try.

Easy Data I/O

The tape came promptly, with readable instructions and answers to some questions I had submitted with my order. These questions would have been answered in the instructions, but the MISOSYS folks wrote out the answers anyway.

I had some trouble loading the tape; I tried to load it with SYSTEM in Disk BASIC and it wouldn't load. I assumed that DISKMOD was the program's identifier, but a little reading showed that it loads as DSKMOD. Once I got it loaded, it always produced disk errors during execution. I put the program on disk with TAPEDISK, as suggested in the MISOSYS directions. When the program ran from disk, it worked fine, picking up my copy of the EDTASM and putting it on disk.

Since I've had the program on disk, I've enjoyed using it very much. It permits easy input and output of data (assembly text or object code) to either tape or disk. Disk*Mod provides prompts where filespecs are required. Anyone familiar with EDTASM will find this easy to use.

There are some nice additions to these features. While in the program, you can get a disk directory which shows the memory usage of each disk file, and you can kill files on the disk. You can see how much memory the current program is using, as well as how much text buffer is left. When you exit the program, you can go to DOS automatically, or you can specify any destination address.

Debugging

When reentering the program, you can enter a hex address to protect an area in high memory where there might be a printer driver or other program.

By entering a 0 you can get back into the program, without destroying what was in the buffer when the program was exited, as long as other operations did not overwrite the buffer area while out of the program. This feature makes debugging nearly painless, since you can save source and object codes on disk: Exit, use DEBUG and come back. If when working with the program DEBUG did not overwrite the butter, you're back in business. If the buffer was ruined, it can be emptied and the saved program entered from disk. Tape users will find this quite different from the sequence required for debugging with a tape-based system.

I found only one problem with the adapted program. It doesn't handle some assembly text errors well. When I had a statement that wasn't in correct format, that line and the next several lines printed erratically and illegibly, but the error wasn't pointed out. If the print during assembly starts to look strange, look for errors and you can fix the program and the print at the same time. ■

Execute from Within

You may be wondering if you can execute the *SAVE/RUN command from within the program. It is possible, but it will be the last statement executed in the program, and you must use caution. If you replace the STOP in line 40 of the above program with *SAVE/RUN"PHONE", as long as the recorder has been properly preset to record, the program will record on tape just as if the statement had been typed from the keyboard. However, you will be unable to CONTinue the program. GOTO 50 will work OK (don't RUN 50 as that would clear the variables!). Another point to note is that in certain circumstances a colon may be required before the asterisk, for example, the statement:

```
IF X = 10 THEN *SAVE/RUN"PROGRM"
```

will generate a syntax error, but:

```
IF X = 10 THEN : *SAVE/RUN"PROGRM"
```

will work just fine.

Another feature of KEEPIT is the machine code monitor. Typing:

```
*OPEN"NNNN"
```

where NNNN is replaced by an address in hexadecimal, displays 16 bytes of memory beginning with the specified address. The bytes are displayed in hex with their associated ASCII characters displayed on the next line.

Typing a two-digit hex code will change the leftmost byte of the series; the entire series is then incremented by one. The series of 16 bytes can be incremented or decremented one byte at a time, without changing any bytes, by holding down the left or right arrow keys. You can move 16 bytes at a time by using the up or down arrow keys. Lest you get confused, the address of the leftmost byte of the series will always be displayed in the upper left-hand corner of the video display.

Once you have typed a machine language program using the monitor, you may BREAK and seve it to tape by using the command:

```
*SAVE/OPEN "PROGRM,NNNN,NNNN,NNNN"
```

where PROGRM is replaced by the program file name and the NNNN's are replaced by the start, end and entry points of the block of machine code to be saved. The resulting tape can be loaded under the SYSTEM command.

The final command, *NEW, restores a BASIC program that has been wiped out accidentally by typing NEW.

KEEPIT is supplied on cassette with both the SYSTEM (object) program and the editor-assembler format source code. This is not the usual case with programs of this type but should be encouraged, since it makes user modification of the program much easier, and also makes it possible to relocate the program anywhere in memory. (KEEPIT is normally supplied with 4K, 16K, 32K, and 48K versions on the tape—these will load at the top of available memory.)

I was unhappy with the delay loop for the KBEEPFIX routine auto-repeat feature. I tend to leave my fingers resting on the keys, which resulted in unwanted repeats of the last key depressed. With the source code provided, I was able to lengthen the delay to an amount of time more to my liking.

KEEPIT is compatible with other special-command routines (such as the Exatron Stringy-Floppy routines). If you don't have a disk system, KEEPIT may prove to be one of the most useful utilities you own, especially if you use it to eliminate some of those time-consuming PRINT #–1 and INPUT #–1 statements.■

ISAR (Information Storage and Retrieval)
The Alternate Source (TAS)
Lansing MI
$17 on disk

by R. Louis Zeppa

For my out-of-print book business I needed an inexpensive method that would help me create book lists. Compiling catalogs by hand and typewriter is slow and prone to error. By this method books are listed on 3×5 cards, sorted, and then the list is typed—a minimum of three steps for each book. Catalogs of 200 to 300 books have taken up to a month to prepare.

With ISAR, I completed two separate 500+ entry catalogs in three days each. First I entered the book citations directly and then let the computer sort and type the catalogs.

ISAR is made up of modules, of which there are ten. Module 1 is the driver or menu module. The basic ISAR includes six more modules: create a file (2), add records (3), change or delete records (4), sort (5), screen scan or search (6), and format hardcopy reports (7). All processing focuses through module 1, that is, you cannot add records and then jump directly to change records. The file name must be entered each time you pass through module 1, thus it is possible to enter the file name more times than is reasonably necessary.

ISAR does have some problems. It cannot add, delete, or modify entries during the same pass through a file. ISAR's sort is frustrating because its printout formats entries but not pages.

Taken singly, none of these is a major deficiency. However, these little quirks can be irritating. For example, you must make two passes through a file to delete and change records. This is because it is the same module and after, for example, deleting, instead of bouncing back to ask if you want to change some records in the same file, ISAR asks if there is another file you wish to delete items from.

Another limitation is that ISAR won't do a multiple field sort. Yet with proper coding and planning it can be simulated. For example, sorting a mail list first by name and then by zip code will produce a list sorted alphabetically by name within each zip code.

There is yet another shortcoming with ISAR—it has an in-memory sort. A file with many records will overwhelm memory if the field being sorted is longer than 18 characters. Although the sort is fast, a slower disk sort would be more flexible.

When reports are formatted for hardcopy, you must sort out deleted items or they show up as skipped lines in the printout. There are more hardcopy problems, but space must cut this critical list short.

You must bend, beat, and squeeze your work into ISAR's limitations. If the manual was accurate and referred to specific lines in the modules, ISAR could be relatively easily modified to suit your own needs. It would still be limited, however.

TAS is committed to improving ISAR with new modules. Extensions to ISAR which are set up in the same way as the add-change-delete functions will continue to hinder its easy usefulness.

For the occasional user who won't do a lot of file manipulation, ISAR will be very useful, at a good price and, to be fair, ISAR is not advertised as a solution to business or bibliographic problems. It was "originally designed to provide personal users with a low cost data management system." As critical as I am, it has proven valuable, but for large and potentially complicated files, ISAR is too difficult. In this case, my advice would be: Spend more money for a more flexible program.■

When DecisionMaster™ speaks everybody listens.

Let's face it. We all have to make decisions. Decisions that can change our lives. Decisions that can make us happy or unhappy. Decisions that could win us fame or fortune. Now, DecisionMaster can help you make the best decisions of your life.

Use Bayesian theory to peer into the future... even if you've never heard of the Bayes' Rule. Develop complete weighted factor analysis... without knowing what one is. Use discounted cash flow to compare investment alternatives without bothering with present value tables. These and other sophisticated theories that were once the exclusive domain of professors and top business executives are *built into* DecisionMaster's algorithms... so you can use them at the touch of a key!

DecisionMaster is easy to use. It features:

- A fully documented manual developed by an authority in the field.
- A unique, screen-controlled cross reference system.
- A powerful formatted-screen data entry system.

You'll use DecisionMaster in hundreds of routine decisions, as well as more important ones such as • Buying a house • Changing jobs • Selecting investment • Evaluating insurance policies • Expanding product lines • Leasing vs. purchasing.

If you buy only one computer program this year, make it DecisionMaster. And when it speaks, *listen.*

DECISION? MASTER™

#302

To order, see your software dealer or return this form with your check to: DecisionMaster.
Dept. 80 . 10428 Westpark . Houston, Texas 77042 Add $1.50 for shipping and handling

Diskette for: ☐ Apple II* (32K) $29.95 ☐ TRS-80** (Level II-32K) $29.95 ☐ TRS-80** (Model II) $39.95

Name

Address

City State Zip

CHARGE by phone toll-free: **1-800-231-5768 Ext. 306**
(In Texas: 1-800-392-2348) or return this form:
☐ VISA ☐ MasterCard Bank Code

Account No.

Signature

Expiration Date

* Apple II is a registered trademark of Apple Computers, Inc.
** TRS-80 is a trademark of the Radio Shack Division of Tandy Corp.

These Next 4 Pages are for TRS-80* Owners ONLY!

The next 4 pages contain over 100 programs for your TRS-80. Whatever your interests, we have a software program for you. We list sections on Home/Personal, Business, Games, the Arts, Home Education, Utilities, Special Business, Flight Simulations, Electronics, Comp-U-Novels, and Popular Games. These programs can be purchased through your local Instant Software dealer, or you can call us directly using our toll free number. We ship our orders the same day we receive them. Browse through these 4 pages, we're sure you'll enjoy your selections. Remember: WE GUARANTEE IT!

UTILITIES

TRS-80 UTILITY I—Give your program that professional look. RENUM: Renumber any Level II program to make room for modification or to clean up the listing. DUPLIK: With this program you can duplicate any BASIC, assembly/machine language program, verify the data and record the program to tape. You can even record Level I programs on a Level II keyboard. **(T1) Order No. 0081R $9.95.**

TRS-80 UTILITY II—Change the drudgery of editing your programs into a quick, easy job. It includes: • CFETCH: You'll be able to merge consecutively numbered BASIC programs into one program. It will also search through any Level II program tape and display the file names for all programs. • CWRITE: Combine subroutines that work in different memory locations into one program. It works with BASIC and/or machine-language programs and will give you a general checksum to verify that your program hasn't dropped any bits. **(T1) Order No. 0079R $9.95.**

THE COMMUNICATOR—This package lets you transmit data over the telephone lines. The full ORIGINATE/ANSWER capability allows your TRS-80 to be controlled from a remote-based terminal, or allows two TRS-80s to "talk" to each other. You can transmit data or programs from home base to a remote terminal. There will be a simultaneous display of information on both video monitors. Requires a modem and RS-232 interface for each terminal. **(T1) Order No. 0126R $9.95.**

TERMINAL-80—Communicate with the rest of the world! These programs give you control of the RS-232 port of your Expansion Interface. You can connect one or more serial terminals to your TRS-80 and it will accept input from the RS-232 interface just as if it were entered from the keyboard. Your TRS-80 can also be transformed into a dumb terminal, for use in a time-sharing situation to talk with "big" computers via a modem. The LPRINT/LLIST commands will transfer a program to a receiving computer. Supports upper/lowercase, Level II & III control characters, and all functions such as CHR$. The baud rate is software controlled for your convenience. Requires an RS-232 interface. **(T1) Order No. 0130R $24.95.**

DISK SCOPE—Need to check out the contents of a disk? Then check out these three programs. • FILELOC: If you know the name of the program or data file, FILELOC will show you which tracks and sectors contain that file, as well as how much memory the file takes when loaded into RAM. You can then print the information, search for a new file or exit to BASIC. • CDISK: This utility and test program allows you to view any track and sector on your disks in ASCII, Hex and screen POKEs. It disregards all protection codes. • PASSWORD: This machine-language program not only gives you a password for individual files, but for whole disks as well. **(T2) Order No. 0139R $19.95.**

DISK EDITOR—This machine-language program give you total access to ANY byte of information in ANY sector in ANY track of your disk! You can examine, alter, add and delete information with ease. You can even search for a specific string (up to 8 characters long). If you need hardcopy, use the LINEPRINT command to send a copy of the video display to your printer. It can be used with TRSDOS, NEWDOS and MicroDOS. Both the 35 and 40 track versions are included. **(T2) Order No. 0160RD $39.95.**

BPA (BASIC PROGRAMMING ASSISTANT)—BPA does three things for you: (1) It will list the variables used in a BASIC program. Optionally, it will list the line numbers where each variable appears; the variable-type symbol (string, integer, single or double precision); whether it is dimensioned and where it is changed. (2) It will produce a cross-referenced list of line numbers for GOTO's, GOSUB's and IF...THEN statements. (3) It will list the line numbers where a selected BASIC function word (e.g., INPUT, PRINT) is used. **(T1) Order No. 0203R $14.95.**

TLDIS & DLDIS—These two utilities are ideal for those who wish to decipher and/or modify machine-code programs. TLDIS (Tape-based Labeling DISassembler) and DLDIS (Disk-based Labeling DISassembler) are three-pass, label-assigning disassemblers that assign labels (where appropriate) to the routines in a machine-language program. Their output is almost identical to that of a hand-assembled source code. TLDIS can send the disassembly to cassette tape, DLDIS can send it to disk; both send it to the video monitor. Each version can be reassembled using Tandy's EDTASM or Apperat's disk extension of EDTASM, respectively. You can also send either disassembly to a printer (R/S parallel port). Because of the labels, it is a simple matter to change any object code program by disassembling it and making changes to the resulting source code, without losing track of the jump/load addresses. Labels start at "AA00" and increment up, in even numbered steps (AA02, AA04, etc.). The odd numbers (AA01, AA03, etc.) are left for your (optional) use in the reassembly. **TLDIS (T1) Order No. 0230R $14.95. DLDIS (T2) Order No. 0231RD $19.95.**

THE DISASSEMBLER—This is a single-pass, hex-notation that sends its output either to tape or to a lineprinter (R/S parallel port). The tape output is directly compatible with Tandy's EDTASM, so you can disassemble an object code tape and output it to tape, then use EDTASM to add, delete, change and re-assemble your new version. It displays the *displacement* and *absolute address* of any relative jumps made by the disassembled program. It also displays and ASCII characters used in an LD or CP opcode. It is relocatable and you can jump to memory locations and transfer control between Disassembler and other utility programs. **(T1) Order No. 0239R $9.95.**

There are over 300 Instant Software dealers throughout the U.S.A. and the world.

Go see your local Instant Software dealer before Christmas. He has a wide selection of Instant Software.

CODE—Minimum System Required

(T1) = TRS-80 Model I Level II, 16K RAM
(T2) = TRS-80 Model I Level II, 16K RAM with Expansion Interface 16 + K RAM and one disk drive
(T3) = TRS-80 Model II, 32K RAM

SEE YOUR LOCAL INSTANT SOFTWARE DEALER OR

Just Call Toll-Free 1-800-258-5473

Instant Software™

PETERBOROUGH, N.H. 03458

We Guarantee It!

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

THE ARTS

COMPU-CAROLS—We are proud to present a selection of Christmas carols, played by your TRS-80. Just place an AM radio next to your keyboard and you'll be amazed at the quality of this computer-generated music. You'll hear AWAY IN A MANGER, NOEL, SILENT NIGHT, O LITTLE TOWN OF BETHLEHEM and eight more of your favorite carols. **(T1) Order No. 0036R $9.95.**

DOODLES AND DISPLAYS II—It includes: • DOODLE PAD: Draw pictures and save them on cassette tapes. • SYMMETRICS: An electronic kaleidoscope that's constantly changing. • DRAWING: Like DOODLE PAD, but for the serious artist. Over 40 user commands. • RANDOM PATTERN DISPLAY: The computer does the drawing, but those with itchy fingers can make alterations. • MATHCURVES: Bring those geometry lessons to life. Six different geometrical curves on the screen of your TRS-80. • RUGPATTERNS: Designs rug patterns with a choice of user or computer control. **(T1) Order No. 0042R $7.95.**

MUSIC MASTER—Includes these four audio treats: • MICRO-ORGAN: This program changes your computer into a musical instrument, with a range of four octaves with three voices! You can play sharps and flats to imitate the sounds of an organ, harpsichord or piano. • KALEIDOPY: Now you can have a computerized "player piano." Generate a symmetrical graphics pattern and then see it transformed into music. • COMPOSER: Experiment with computer-generated music. You can select the length of the piece, its scale, and its tempo. • KEYMANIA: Test your memory and your musical ear. One to four players try to repeat the melody that the computer creates. **(T1) Order No. 0064R $9.95.**

ELECTRONICS

HAM PACKAGE I—This versatile package lets you solve many of the problems commonly encountered in electronics design, including: • BASIC ELECTRONICS WITH VOLTAGE DIVIDER: Solve problems involving Ohm's Law, voltage dividers and RC time constants; • DIPOLE AND YAGI ANTENNAS: Design antennas easily, without tedious calculations. **(T1) Order No. 0007R $7.95.**

ELECTRONICS I—This package will not only calculate component values for you, it will also draw a schematic diagram. Included are: • TUNED CIRCUITS AND COIL WINDING: Design tuned circuits without restoring to cumbersome tables and calculations; • 555 TIMER CIRCUITS: Design astable or monostable timing circuits using this popular IC; • LM-381 PREAMP DESIGN: Design IC preamps with this low-noise IC audio amp. **(T1) Order No. 0008R $7.95.**

QSL MANAGER—Ever looked at your log book and wondered if you sent a QSL card to the operator you worked last week? Maybe you sent a QSL but can't remembered getting one in return. The QSL MANAGER will help you set up a computerized log book that gives you *instant access* to your records. Make complete log entries which include: Date, Time, Call sign, Name, Band, both the sent and received Signal Reports, the Mode, whether a QSL card was sent or received and any remarks you want to add. The QSL MANAGER program has built-in editing features that let you keep your log book up to date. **(T2) Order No. 0151RD $19.95.**

HOME EDUCATION

MONEY MADNESS—You can experience the Raw Power of High Finance with two Big Money empires. • MILLIONAIRES: Can you manipulate $1000 into a million dollars in fifteen years? It all depends on your strategy as you buy and sell properties, negotiate bank loans, collect rentals and accept sealed bids. • TIMBER BARON: An in-depth experience of the timber business, from the time you cut the trees until your milled lumber reaches the market. These transections are affected by those tough, unexpected eventualities that can upset the most careful plans. **(T1) Order No. 0156R $9.95.**

TEACHER'S AIDE—Now you can have the benefits of Computer Aided Instruction (CAI) in your own home. Create a question and answer lesson (up to 8000 characters), save the lesson on disk, then create an entire sequence of lessons. Perfect for parents, teachers and students who need the unlimited patience and undivided attention only a computer can provide. **(T2) Order No. 0214RD $34.95.**

GRADE BOOK—Teachers, now you can use the speed and accuracy of the computer to help calculate student grades. Just type in the grades for tests, quizzes, homework, classwork or special projects to calculate and display individual grade averages. You can also obtain a cumulative grade for a specific marking period—or a whole year! **(T1) Order No. 0050R $9.95.**

TEACHER—This program enables you to create your own tests, quizzes and exercises for the education of your children. You can even provide "graphic" reward for your children and provide hints for problem solving. **(T1) Order No. 0065R $9.95.**

LIFE—Create "living" organisms in which cells are constantly active. They are born, they multiply, they die. This computerized version of LIFE is based on the well known game popularized by Martin Gardner. You can create one-cell organisms, then observe their growth patterns. The library of commands give you unlimited versatility in the control of the cell patterns *you* have arranged. **(T1) Order No. 0078R $9.95.**

ARCHIMEDES' APPRENTICE—This two-part package will teach you the formulas used to find the volume of any solid object including paralellopipeds (cubes and rectangular solids), prisms, pyramids, cylinders, cones and spheres. It will show you on-screen diagrams of these figures, and present you with the formulas you'll need to compute their volumes. **(T1) Order No. 0092R $9.95.**

TYPING TEACHER—This complete seven-part package takes you from initial familiarization with the keys, through typing words and phrases, to complete mastery of the keyboard. Your computer can even become a bottomless page for typing practice. **(T1) Order No. 0099R $9.95.**

VIDEO SPEED READING TRAINER—Most people's reading speed is limited simply because they read individual letters or words. Now you can increase your reading speed and comprehension by reading whole words and phrases. This package will train your mind to quickly recognize numbers, words, letters and phrases. Start at any speed level at which you are comfortable and the computer will automatically advance you as your reading speed and comprehension increases. **(T1) Order No. 0100R $9.95.**

WORDWATCH—four different programs to entertain and educate. • WORD RACE—race to the finish line of defining words correctly; • HIDE N SPELL—find the misspelled word, then correct it; • SPELLING TUTOR—a spelling lesson, but beware, the spelling may become unusual. There you have it. Wordplay × four = WORDWATCH. **(T1) Order No. 0111R $7.95.**

MIND WARP—This game includes. • MIND TWIST: a Mastermind-type game with a twist. Try to guess the computer's secret digit sequence. • MIND BENDER: A multi-level game where you must discover the computer's secret code. It's no mystery, the MIND WARP package is for puzzle lovers everywhere. **(T1) Order No. 0118R $9.95.**

INVESTOR'S PARADISE—Here are two programs to test your skill in the stock market. • STOCK TREK: a stock market simulation in which you and up to five other investors buy and sell stocks. • SPECULATION: a step beyond a mere simulation, you enter financial data on up to 25 real companies and start playing the market. This package lets you experience the thrills and triumphs of the stock market without risking a dime! **(T1) Order No. 0125R $9.95.**

IQ TEST—IQ TEST will administer and score an intelligence test in just 30 minutes. There are three equivalent tests, each consisting of 3 questions that survey your general knowledge and problem solving abilities. **(T1) Order No. 0157R $9.95.**

SPECIAL BUSINESS

BOWLING LEAGUE SECRETARY—This package is simple to operate and provides a dynamic reference to all the names of individual bowlers, their team numbers, scores, team names, league data and all necessary statistics. The system is highly adaptable, with 17 different scoring options that allow you to custom tailor the program to suit your league's special needs. And, if you even have any problems, simply type HELP and the program will give you an explanation of what information is needed—complete with a sample entry. The system puts at your fingertips all individual weekly scores, team cumulative scores, bowler cumulative scores and individual leaders in the following categories: high single, high series, high average and high points. **(T2) Order No. 0095RD $49.95.**

SEE YOUR LOCAL
INSTANT SOFTWARE DEALER OR

Just Call Toll-Free
1-800-258-5473

BEGINNER'S RUSSIAN—In order to understand a foreign culture, you must know its language. The three programs in this package will give you on-screen displays of the characters of the Cyrillic alphabet, detailed instructions of their proper pronunciation and exercises that will have you recognizing and speaking simple Russian words. An excellent package for students, businessmen, scientists or anyone who is interested in learning the Russian language. **(T1) Order No. 0136R $9.95.**

EVERYDAY RUSSIAN—will acquaint you with the words for various foods, places to eat, signs and the names of stores—exactly what a traveller needs to know. Each of the three parts of the package not only teaches you the words but quizzes you on them as well. You can even practice typing in Russian. Discover the Russian language today! **(T1) Order No. 0137R $9.95.**

SEE OUR ADS ON PAGES 196 & 197 FOR ALL NEW INSTANT SOFTWARE PROGRAMS

BOWLING LEAGUE STATISTICS SYSTEM—Keeps a computerized list of league data, team data and data for each bowler. Extremely flexible, it has a total of 16 different options to let you modify the program to suit your league's rules. It is easy to use and has a built-in "HELP" feature to aid you. **(T1) Order No. 0056R $24.95.**

HOME/PERSONAL

HOUSEHOLD ACCOUNTANT—Save with these two programs: • BUDGET & EXPENSE ANALYSIS: It has nine sections for income and expenses and an option for quarterly/yearly reviews. • LIFE INSURANCE COST COMPARISON: Compare the total costs of various insurance policies. Contrast term with whole life. It will store and display up to six prospective policies. **(T1) Order No. 0069 $7.95.**

PERSONAL BILL PAYING—You can keep a computerized list of ALL your bills (up to 22 accounts), each listed with its name, number, due date and amount owed. Individual accounts can be displayed with a month-by-month breakdown of payments (including check numbers) and current accounts can be seperated from inactive ones. It allows you to save the data to tape for future use. **(T1) Order No. 0103R $7.95**

WRITE FOR OUR NEW INSTANT SOFTWARE CATALOG

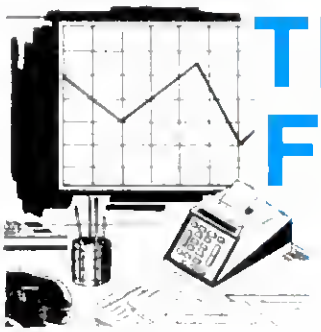
We Guarantee It!

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

Instant Software™

PETERBOROUGH, N.H. 03458



TRS-80* Software From Function to Fantasy

POPULAR GAMES

BEGINNER'S BACKGAMMON/KENO—Why sit alone when you can play these fascinating games: • BACKGAMMON: Play against the computer in a game that's sure to sharpen your skills; • KENO: Enjoy this popular Las Vegas gambling game—guess the right numbers and win big! **(T1) Order No. 0004R $7.95.**

CHESSMATE-80—This versatile chess opponent gives you a choice of ten levels of play, from the "blitz" level (the computer has 3 seconds to move) to the infinity level (where the computer will consider every possible move—which could take years). This machine-language program is a conservative player and follows all the rules of international play. CHESSMATE-80 can teach you how to move and allow you to set up the board and play end games or special problems. CHESSMATE-80 battled Sargon II to a draw at two minutes a move and beat Microchess 1.5 in six moves. **(T1) Order No. 0057R $19.95.**

YOUR CRIBBAGE AND CHECKERS PARTNER—CRIBBAGE is a two-person game that you are sure to enjoy. This is NOT a tutorial—it is a game worthy adversary. CHECKERS: An old favorite which follows international rules, including multiple jumps. **(T1) Order No. 0068R $9.95.**

CARDS—A one-player package to let you play, with your computer, these famous games: • DRAW AND STUD POKER: These programs will keep your game sharp; • NO-TRUMP BRIDGE: Develop your strategy and (hopefully) increase your skill. **(T1) Order No. 0063R $7.95.**

FLIGHT SIMULATIONS

RAMROM PATROL/TIE FIGHTER/KLINGON CAPTURE—• RAMROM PATROL: Destroy the RamRom ships before they capture you. • TIE FIGHTER: Wipe out the enemy Tie fighters and become a hero of the Rebellion. • KLINGON CAPTURE: You must capture the Klingon ship intact. **(T1) Order No. 0028R $7.95.**

FLIGHT PATH—This three-part package includes: • MOUNTAIN PILOT: Become a daring bush pilot and fly supplies to a remote mining camp. You must cross mountain ranges and struggle with headwinds, tricky navigation and rapidly diminishing fuel. • O'HARE: A control tower simulation for you would-be Air Traffice Controllers. You are responsible for the lives of hundreds of passengers as you guide aircraft through your control sector. • PRECISION APPROACH RADAR: Combines the skills of pilot and Air Traffic Controller, as your commands guide an aircraft in its approach to the field and a safe landing. **(T1) Order No. 0171R $9.95.**

BALL TURRET GUNNER—Imagine yourself at the control console of a strategic laser weapon, deep in the space lanes. Your hindsight detector informs you of a Gnat fighter coming in for an attack so you swivel you laser turret until you can see the target. Watch the Range Indicator and your Targeting Computer's readout closely, because you'll only have a fraction of a second to catch him in your sights. Will you transform the Gnat into a ball of ionized gas or will you see that blinding flash that means The Big Demotion? BALL TURRET GUNNER, with you choice of multiple levels of difficulty, optional sound effects and excellent graphics, is more than a game. It's an event to be savored. **(T1) Order No. 0051R $9.95.**

JET FIGHTER PILOT—In this brilliantly realistic simulation, you become the pilot of a twin turbo-jet fighter. Begin your mission from either the deck of a carrier or from an airfield. During flight, you'll need to constantly monitor your display and make the necessary adjustments to the throttle, flaps, and air spoilers; you must decide when to retract landing gear and release your drop tanks! There is an on-board Navigational Computer, a Glideslope/Localizer and a Weapons Control Computer. Earn your wings with JET FIGHTER PILOT. **(T1) Order No. 0159R $14.95.**

SPACE TREK II—Protect the quadrant from the invading Klingon warships. The Enterprise is equipped with phasers, photon torpedoes, impulse power and warp drive. **(T1) Order No. 0002R $7.95.**

AIR FLIGHT SIMULATION—Take off and land your aircraft without making a crater. This "instruments only" simulation starts you with a full tank of fuel, which gives you a maximum range of about 50 miles. You'll get constant updates of air speed, compass heading and altitude. After you've acquired a few hours of flight time, you can try flying a course against a map or doing aerobatic maneuvers. **T(1) Order No. 0017R: $9.95.**

SPACE TREK IV—• STELLAR WARS: Engage and destroy Tie fighters in your attack on the Death Star. For one player. • POPULATION SIMULATION: A two-player game where you control the economy of two neighboring planets. You must decide: Guns or Butter? **(T1) Order No. 0034R $7.95.**

BASIC AND INTERMEDIATE LUNAR LANDER—Bring your lander in under manual control. The basic version is for beginners; the intermediate version is more difficult, with a choice of landing areas and rugged terrain. **(T1) Order No. 0001R $7.95.**

COSMIC PATROL—We put you in command of a small interstellar patrol craft. You must defend Terran space and prey on the Qualon freighters that carry vital war supplies—but beware of their I-Fighter escorts. They're well armed, extremely fast and they NEVER miss! With its real-time action, impressive sound option and superb graphics, this machine-language program is the best of the genre. **(T1) Order No. 0223R $14.95.**

Airmail Pilot —Return to the early days of aviation. You must fly the mail from Columbus to Chicago. Your Jenny, a cloth-covered biplane, must take you through unpredictable winds, hail and electrical storms. Your mission is to get the mail through in the shortest possible time. There is an on-board clock to time you flight, from takeoff to touchdown. . . assuming you are able to complete it. **(T1) Order No. 0106R $9.95.**

NIGHT FLIGHT—Your mission is to fly over the North Atlantic and make a nighttime photo/recon flight above the enemy fleet. NIGHT FLIGHT lets you take-off, fly and land a propellar-driven aircraft. You can practice approaches and landings with an on-screen display of the landing field information—it will practically teach you to fly. **(T1) Order No. 0117R $9.95.**

COMP-U-NOVELS

WHO-DUN-IT? Criminal elements have committed five dastardly crimes. As the investigating detective, you must solve them.

You can compete against either Detective Nybbles, a computerized sleuth, or up to four other human detectives.

• DEDUCTION: Guess the order of four symbols out of six or seven different ones. To make things even more complicated, you can let the computer repeat symbols and have a range of 2401 possibilities. **(T1) Order No. 0047R $7.95.**

SANTA PARAVIA AND FIUMACCIO Become the ruler of a medieval city-state as you struggle to create a kingdom. Up to six players can compete to see who will become the King or Queen first. **(T1) Order No. 0043R $7.95.**

There are over 300 Instant Software dealers throughout the U.S.A and the world.

We ship the same day we receive your order.

CODE—Minimum System Required

(T1) = TRS-80 Model I Level II, 16K RAM
(T2) = TRS-80 Model I Level II, 16K RAM with Expansion Interface 16 + K RAM and one disk drive
(T3) = TRS-80 Model II, 32K RAM

*A trademark of Tandy Corporation

SEE YOUR LOCAL INSTANT SOFTWARE DEALER OR

Just Call Toll-Free 1-800-258-5473

WRITE FOR OUR NEW INSTANT SOFTWARE CATALOG

We Guarantee It!

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS. A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

Instant Software™

PETERBOROUGH, N.H. 03458

HOME/PERSONAL

THE WORDSLINGER—An economical word processing program that was designed for the individual user or small business featuring: automatic formatting; text editing; and tape storage. Once you've used the WORDSLINGER, you won't want to go back to your typewriter. **(T1) Order No. 0129R $29.95.**

MIMIC—Test your memory and reflexes with five versions of this popular game. You must match the sequence and location of symbols displayed on your monitor within the time limit. Instructions on how to produce accompanying sound effects. **(T1) Order No. 0066R $7.95**

CLIMATE COMP—This two-program package includes: WEATHER FORECASTER, which gives you a short range weather forecast based on the information that you enter and WEATHER PLOT, which will display climatological data for any major city in the United States. **(T1) Order No. 0102R-1 $19.95.**

BODY BUDDY—Includes these three programs: ● ADULT CALORIC REQUIREMENTS: Will determine your Basal Metabolic Rate and suggest strategies to achieve your ideal weight! ● FLEXI-DIET: Creates an "infinite" number of diet menus, on a day-to-day basis. Choose your caloric intake, from 600 to 2400 calories per day. The ● ANATOMY QUIZ program teaches a mini-lesson on the various organs of the human body, giving location, size and function(s). **(T1) Order No. 0109R $9.95.**

ENERGY CONSUMPTION—This program will record and analyze your utility bills for up to five years, when you supply the following information. Gas/Water/Electricity used and their respective costs. It will calculate six monthly usage averages and unit costs. Data can be compared for any month or multi-month periods. **(T1) Order No. 0132R $9.95.**

BUSINESS

SALES ANALYSIS—If your business is sales, you're faced with some unique problems. This package is divided into several modules to help solve those problems: The SALES ANALYSIS module is designed to provide guidelines for determining sales performance, to analyze this performance and show you where it can be improved. The DATA STORAGE module allows you to store data in an automated processing ledger. The MANAGEMENT ANALYSIS module can take all the sales records for your group and show you who your best salespersons are, who needs more training and give you a sales forecast. Finally, the MARKET ANALYSIS module can show you where determined sales efforts can produce the most success. **(T1) Order No. 0131R $24.95.**

ORACLE-80—will provide you with business analysis and forecasting capabilities previously available only on large computer and time-sharing systems. A flexible, professional time series analysis and forecasting package for use in product planning, business planning, sales forecasting and more. Financial managers and economists can analyze economic climates and investigate business cycles. ORACLE-80 is designed to be used and understood by the typical businessperson. All input and output is written in plain English and the package documentation carefully explains all the functions of the program. ORACLE-80 puts the future in your hands. **(T2) Order No. 0140R $75.00.**

BUSINESS PACKAGE IV—This business package contains two programs: ● BUSINESS CYCLE ANALYSIS: This program can plot the expansion and contraction cycles of any aspect of your business. ● FINANCIAL ANALYSIS: Now you can get the figures for any type of annuity, sinking fund, or mortgage and compute the yield and value for bonds. The package includes a blank data tape. **(T1) Order No. 0019R $9.95.**

FINANCIAL ASSISTANT—Compute the figures for a wide variety of business needs, including: ● DEPRECIATION: Figure depreciation on equipment five different ways. ● LOAN AMORTIZATION: Enter a few essential factors and get a complete breakdown of all costs and schedules of payment for any loan. ● FINANCIER: Performs thirteen common financial calculations. ● 1% FORECASTING: Use it to forecast sales, expenses, or any other historical data series. **(T2) Order No. 0072R $7.95.**

CHECK MANAGEMENT SYSTEM—Use this program for writing checks and maintaining records. You can make entries, edit/correct entries and print out the checks. It will also search and display records by number, code, date, description or amount. A Code and Search routine allows you to print a report of all checks written for specific expenses. You can print your letterhead and account number at the top of each report. System requirements: **(T2)** with a compatible tractor-feed printer. **0147RD $39.95.**

ACCOUNTS RECEIVABLE/ACCOUNTS PAYABLE—These Model I programs will handle the drudgery involved in AR/AP entries. They will also provide invoices, statements, reports and more. Each program is capable of handling up to 1500 entries per month, posted to as many as 760 accounts. The AR/AP package is ideal for any small business and can easily be used by anyone familiar with AR/AP operations. System requirements (in addition to **T2**: Three disk drives and a Line Printer (tractor-feed). **Order No. 0075RD $199.85.**

MAIL/LIST—With a five-inch drive, you can store up to 600 names per disk without DOS, or 300 names with DOS. The program maintains separate alphabetical and ZIP code files under constant sort. When you add a name or ZIP code to your list, it will be inserted into its correct position in the file. The program will record your data in nine fields: address, city, state, ZIP code, phone number, phone extension and name (2) plus a five character code field. The best feature of this program is the sort process that lets you determine alphabetical or ZIP code order for label printing. **(T2) Order No. 5000RQ $99.00**

ONE-D MAILING LIST—A comprehensive mailing list program that will run on only ONE disk drive! Up to 17 fields of selection for name/address retrieval. Its features include: Auto-sort (alphabetic or ZIP code). Easy error correction and recovery. Prints selective listings. Supports up to 4 drives. Prints mailing labels and listing of all names on file. **(T2) Order No. 0123RD $24.95.**

EXECUTIVE EXPENSE REPORT GENERATOR—Provides you with emergency relief in the form of a clear, plausible expense layout. Input your grand total and cash advance (if any), and you'll receive an itemized expense report, from breakfast to snacks. **(T1) Order No. 0135R $9.95.**

GAMES

WINNER'S DELIGHT—Do you enjoy a challenge? Then try WINNER'S DELIGHT including: ● AMAZING: You must escape from a maze, one that you view from the inside, working against the clock; ● JUNIOR CHECKERS: Not your usual game of checkers . . . the challenge is to beat the computer in the fewest number of moves; ● JUMBO JIGSAW: Fit the pieces together in the fewest number of tries; ● THIRTEEN WAYS: Try to fill up your columns with the numbers you roll on the dice—the computer will try to fill its columns first! **(T1) Order No. 0124R $9.95.**

FUN PACKAGE I—Why call it "Fun Package"? Judge for yourself! This entertaining package includes: •ROCKET PILOT: Flying it is easy—it's the landing that's tough! •PAPER, ROCK, SCISSORS: It's the time-honored game just as you remember it, played against your TRS-80. •HEX I: Just when you master this puzzle game, the computer will increase the difficulty. •MISSILE ATTACK: **Use your missiles to protect your city from jet attack. Requires a TRS-80 Level I 16K. Order No. 0037R $7.95.**

DEMO III—The biggest package ISI has ever released, including: ● RACE 1: Careen around the race course as you try to beat the clock; ● TARGET UFO: Destroy all the invading UFOs; ● LIFE: Experiment with this simulation of the life cycle of a colony of bacteria; ● PHONE NUMBER CONVERTER: Change those hard to remember 7-digit phone numbers into easily remembered words; ● BIORHYTHM: Plot biorhythm curves for anyone, anytime; ● GRAPHICS PROGRAM: This program will show you what your TRS-80's graphics display can do; ● RACE 2: Five different tracks for the more experienced driver; ● HORSE RACE: Up to nine players can bet on and enjoy our most entertaining horse race program; ● DRAWING BOARD: Draw pictures or messages and store them in memory or on cassette tape with this easy-to-use program; ● 24-HOUR CLOCK: Transform your computer into an accurate digital clock. **(T1) Order No. 005SR $7.95**

OIL TYCOON—Avoid oil spills, blowouts and dry wells as you battle to become the world's richest oil tycoon. Two players become the owners of competing oil companies as they search for oil and control their companies. **(T1) Order No. 0023R $7.95.**

BOWLING—Let your TRS-80 set up the pins and keep score. One player can pick up spares and get strikes. **(T1) Order No. 0033R $7.95.**

DEMO II—contains: ● TIC-TAC-TOE: An old time favorite with three levels of difficulty; ● TIME TRIALS: Try to beat the clock as you race your car through curves, chutes, and chicanes; ● MAZE: One or two players can search through the maze for the secret square; ● HANGMAN: One or two players can try to guess the secret word; ● WHEEL OF FORTUNE: Choose your number, place your bet and see if you can break the bank (for one to eight players); ● HURRICANE: You can track and monitor hurricanes in any part of the world; ● BUGSY: Can you build your Z-80 bug before the computer does? ● HORSE RACE: Pick a sure winner and place your bet (for 1 to 100 players). **(T1) Order No. 0049R $7.95.**

BATTLEGROUND—It is late 1944 and the Allied forces are sweeping toward Berlin. As General in command, you study the map. At your command are tanks, planes, artillery, infantry, engineers, and vehicles. The battle map of your sector will fill with markers to show the development of your forces. You and your opponent will assume the roles of warring Generals, as the battle unfolds. The stark reality of World War II comes alive in BATTLEGROUND. **(T1) Order No. 0141R $9.95.**

SKIRMISH-80—Check out these great games: ● MISSION IMPOSSIBLE: Your objective in this real-time simulation is to drive your tank into a prison courtyard, rescue a jailed prisoner and escape; ● TRAP: A two-player game, in which you must maneuver your opponent into a position where he is hopelessly trapped; ● WIPEOUT: A two-player game in which your mobile gun gets points by destroying as many obstacles as possible, but be careful—some of those obstacles are explosive mines; ● BLOCK-'EM: A two-person competition in which your moving "snake" tries to force your opponent to hit either (1) your trail, (2) his own trail, (3) the boundaries of the field, or (4) any randomly placed barriers. The strategy is, of course, to leave you opponent no safe move. **(T1) Order No. 0070R $9.95.**

POPULAR GAMES

GOLF/CROSS-OUT—Have fun with these exciting one-player games. Included are: •GOLF: You won't need a mashie or putter—or a caddie, for that matter—to enjoy a challenging 18 holes
•CROSS-OUT: Remove all but the center peg in this puzzle, and your neighbors will call you a genius. **(T1) Order No. 0009R $7.95.**

We ship the same day we receive your order.

SEE YOUR LOCAL
INSTANT SOFTWARE DEALER OR

Just Call Toll-Free
1-800-258-5473

We Guarantee It!

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS. A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

SEE OUR ADS ON PAGES 196 & 197 FOR ALL NEW INSTANT SOFTWARE PROGRAMS

Instant Software™

PETERBOROUGH, N.H. 03458

80 APPLICATIONS

by Dennis Kitsz

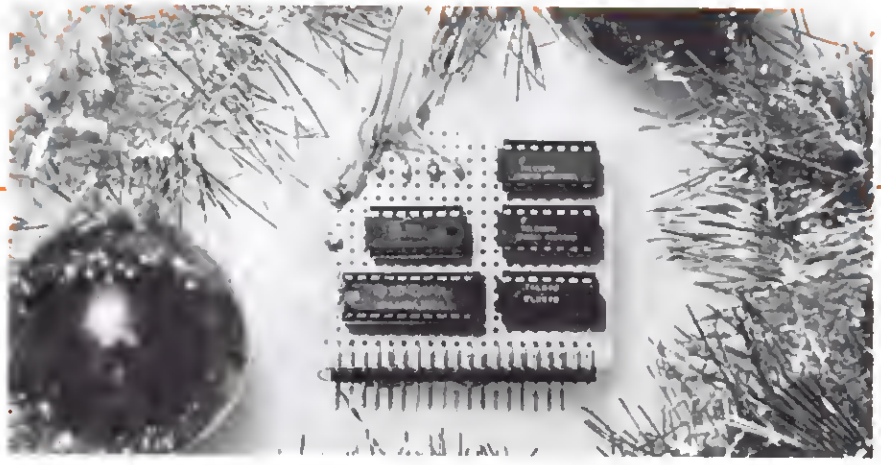


Photo 1. The complete Earie circuit as constructed on a 2 by 2 1/2-inch piece of perf-board.

Now, don't tell me you've never picked up e soldering iron before. Maybe so, but by the time you get done putting together all those easy-to-assemble toys for the whippets, this project will seem like a piece of cake.

So get down to your electronics supply house and get integrated circuits 74LS30 (two), 74LS02, 74LS125, and 74LS374; four 14-pin sockets; one 20-pin socket; five 1K-ohm resistors; a TRS-80 edge connector; some perf-board and a five-volt power eupply kit. Toss in a pair of audio cables, too, end meybe a little box.

What's Kitsz up to this month? Zounds, sound! Not e just e few raspy squawks, but lots of them. . . four separate voices, creeted by e mere 160 bytes of e program! This circuit the "Eerie," is in time for the holidays and inexpensive. A bag of parts would make the great gift for someone to while ewey a chilly January hour or two.

The Circuit

The principle of the hardwere is simple: it merely provides a kind of "window" to e single location in RAM. The location we will be spying on is 4FFF hex (20479 decimal). Z3, Z4, and Z5 create a signal which is activated only when we write to location 4FFF (20479 decimal). See Fig. 1.

Z3 decodes the FF byte of the address. Z5a-c and Z4 decode the 4F byte end combine it with the computer's "write" signal. Z5d NORs the resulting signals together to produce e single pulse defining "write to 4FFF." (See Table 1.)

Z1 acts as an electronic dam end reservoir: Data from the computer confinuously wells up ageinst Z1's input. But the data is permitted to flow into the output, where it is preserved, only when e pulse opens its electronics sluiceway. Unlike circuits where the output status is determined by the stable level of a trigger signal, the 74LS374 lets input flow to output only when the trigger (CLK) signal is changing from zero to one. That is, it is edge triggered rather than level triggered.

The CLK signal for Z1 is the output of Z5d, "write to 4FFF." So when we write to memory location 4FFF, whatever data is being placed in that memory eddress will also be brought to the output of Z1.

Finally, Z2 conteins four separate three-state buffers. A buffer is merely e device which allows a signal to pass through it, unchanged, in one direction. The three-state quality is an important one for computers, because dozens of separate circuit outputs are connected to the same set of wires. Confused signals and damaging short-circuits must be prevented. Thus, not only can some devices output a high signal (1) or low signal (0), but they can also turn invisible when they are not needed. This is the important third state.

This three-stete butter Z2, though, is not part of any complicated data or eddress bus—its outputs only go to some resistors. Why the third state? It allows us to turn the sound off during a rest; the reasons will become clearer when we take e look at the software.

A few resistors complete the circuit, blending the four outputs into two, as well es offering the outputs of Z2 a bit of protection egeinst casual cable connecting. The discrete channels can be used for those with quadriphonic systems.

The Earie is very simple to build, and can be completed in an evening. Remember to use a regulated five-volt power supply. A good experimenter's supply is sold by Jameco Electronics (1021 Howard Avenue, San Carlos, CA 94070, (415) 592-8097) for $14.95, although a simpler source, such as thet shown in Fig. 2, is adequate for the sound circuit.

Any type of wiring can be used, be-

To decode "write to address 4FFF", convert the address to binary, and identify the address lines associated with each bit:

Hex Value:	4				F				F				F			
Bit:	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1
Address Line:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Step 1. Feed eight address lines (0 through 7) to the inputs of an eight-input NAND gate. When all lines go high (1), the output of the NAND gate will be zero.

Step 2. Feed address lines 12 and 13 into a NOR gate. When these swing low, the NOR gate goes high.

Step 3. Feed address line 15 into both inputs of a NOR gate. When this line goes low, the NOR gate goes high.

Step 4. Feed the WRITE signal, which is active low, to both inputs of e NOR gate. When the signal is active, the NOR gate goes high.

Step 5. Feed the outputs of the above three NOR gates, which will be high when they form the values needed, into three inputs of an eight-input NAND gate.

Step 6. Feed the remaining address lines (8, 9, 10, 11, and 14) to the other five inputs of the eight-input NAND gate. When these lines go high together with the lines from Step 5, the output of the gate will be low.

Step 7. Connect the outputs of both eight-input NAND gates to a NOR gate. When both NAND outputs are ective (they will be low), the NOR gate goes high. Only when the address 4FFF appears simultaneous with a WRITE signal will this combined signal go high.

Table 1

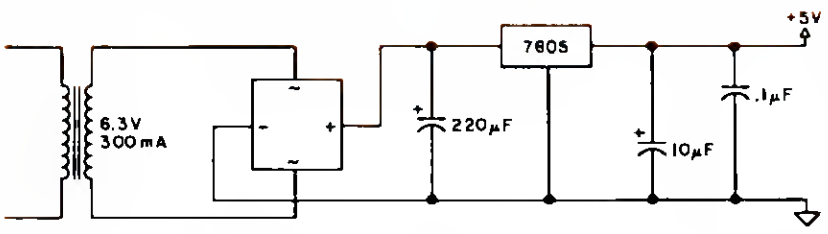


Figure 2. Power source for the sound circuit; any regulated five-volt source is adequate.

cause in this circuit, neatness is a matter of aesthetics rather than necessity. For those new to digital hardware, I particularly recommend the wire-wrapping method as a contribution to sanity; errors in wiring can merely be unwrapped.

Photo 1 shows the completed circuit, which fits on a small 2 by 2 1/2-inch perfboard. The "header" connector on the card's edge is e useful, money-saving substitute for expensive 40-wire cables (which are clumsy to strip and solder to circuit boards). Instead, this connector mates with a cable whose far end plugs into the TRS-80 expansion connector. The cable and a pair of headers can be obtained from Digi-Key Corp., P.O. Box 677, Highway 32 South, Thief River Falls, MN 56701, (800) 346-5144. The cable, which can be used for many projects, costs $11.95; a pair of headers is $3.49.

Making Sound with Software

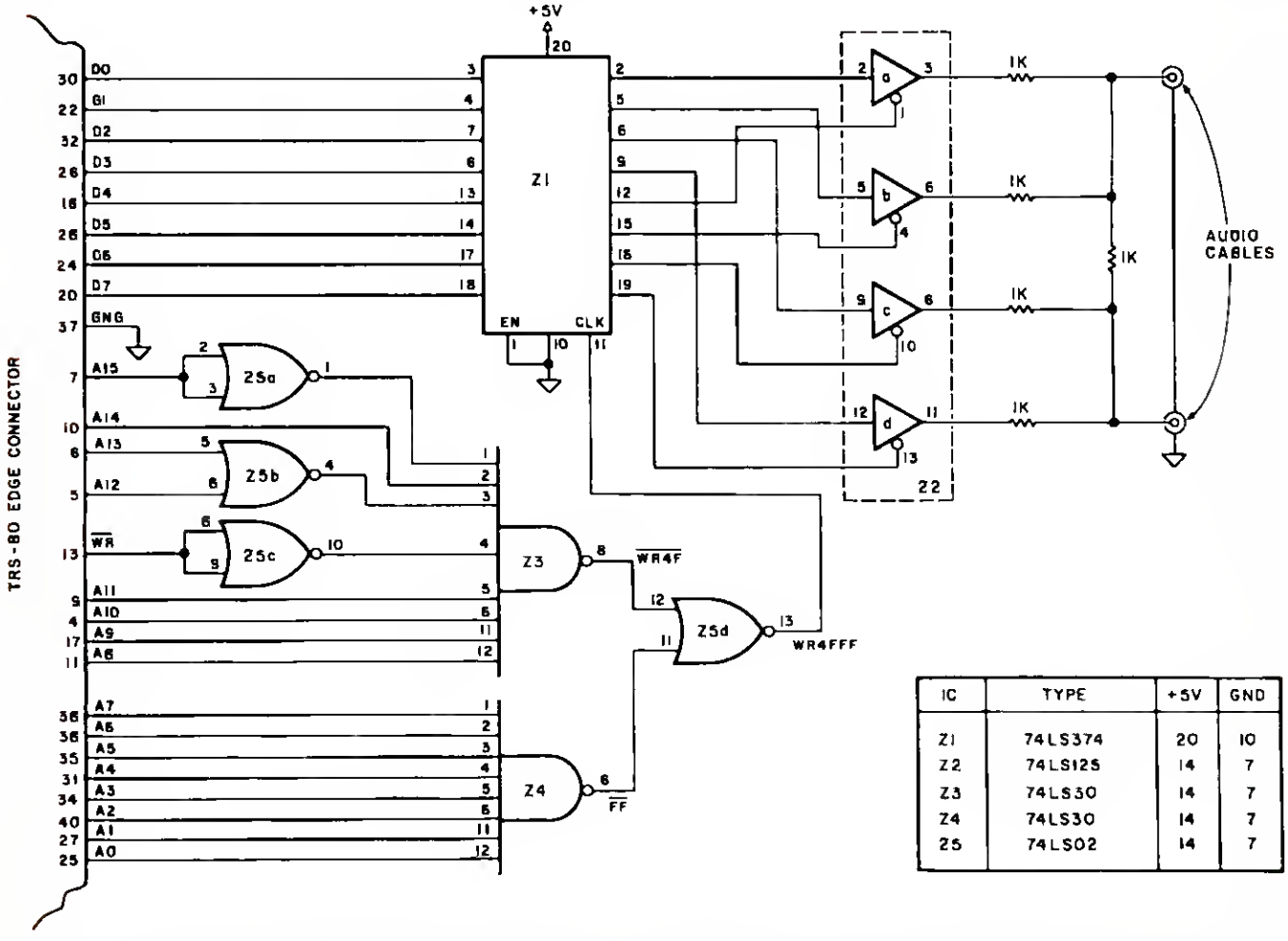
The production of interesting sound and music with microcomputers is a considerable challenge. December *Kilobaud Microcomputing* features more than a half-dozen ways to create music. Some of the newer integrated circuits described can produce three-voice music, but the programming can be complicated.

The way the Earie creates sound is by listening to the activities taking place in memory location 4FFF. In fect the sound is no more than the pattern of changing electrical impulses of various bits being stored in that memory address!

By carefully considering computer timing, we can turn individual bits of that memory address off and on often enough to produce a square wave. There is only one serious problem: time. BASIC is *much* too unwieldly to use for producing multi-voice sound waves because even its simplest instructions take a large fraction of a second to execute. A simple loop like

```
10 FOR X=0 TO 255 : POKE 20480,X : NEXT
```

takes two full seconds to complete. There's not much monophonic music in



IC	TYPE	+5V	GND
Z1	74LS374	20	10
Z2	74LS125	14	7
Z3	74LS30	14	7
Z4	74LS30	14	7
Z5	74LS02	14	7

Figure 1. Complete diagram of the simple circuit. The total cost of parts, including cable, is under $20.

```
                00100 ; ******************************************************
                00110 ;    ELEMENTARY (AND SLOW) FOUR-VOICE MUSIC SUBROUTINE
                00120 ;    BY DENNIS BATHORY KITSZ,  ROXBURY, VERMONT  05669
                00130 ; ******************************************************
                00140 ;
4F60            00150          ORG     4F60H           ;  :NEAR TOP OF MEMORY
4F60 DD21#050   00160          LD      IX,5000H        ;  :START PITCH & RHYTHM
4F64 01FF4F     00170          LD      BC,4FFFH        ;  :MEMORY-MAPPED SOUND
                00180 ;
                00190 ; ******************************************************
                00200 ; OUTER (INTER-NOTE) LOOP BEGINS HERE; T-STATES 212 - 244
                00210 ; ******************************************************
                00220 ;
4F67 D9         00230 LOOP1    EXX                     ;04:READY DURATION REGS.
4F68 DD4600     00240          LD      B,(IX+0)        ;19:MSB OF NOTE DURATION
4F6B DD4E01     00250          LD      C,(IX+1)        ;19:LSB OF NOTE DURATION
4F6E D9         00260          EXX                     ;04:STASH REGISTER AWAY
4F6F DD6602     00270          LD      H,(IX+2)        ;19:FIRST PITCH INTO H
4F72 DD6E03     00280          LD      L,(IX+3)        ;19:SECOND PITCH INTO L
4F75 DD5604     00290          LD      D,(IX+4)        ;19:THIRD PITCH INTO D
4F78 DD5E05     00300          LD      E,(IX+5)        ;19:FOURTH PITCH INTO E
                00310 ;
                00320 ; ******************************************************
                00330 ; EACH VALUE ACQUIRED FROM IX IS TESTED TO SEE IF IT IS 0
                00340 ; AND THE VOICE IS TURNED OFF IF IT IS (DEFINING A REST).
                00350 ; ******************************************************
                00360 ;
4F7B 0A         00370          LD      A,(BC)          ;07:READY TO TWEAK MEM
4F7C E60F       00380          AND     0FH             ;04:TURN ALL VOICES ON
4F7E 24         00390          INC     H               ;04:BUMP VALUE; REST TEST
4F7F 25         00400          DEC     H               ;04:BUMP VALUE; REST TEST
4F80 C2854F     00410          JP      NZ,REST1        ;10:ONLY 00 DEFINES REST
4F83 CBE7       00420          SET     4,A             ;08:SILENCE VOICE IF REST
4F85 2C         00430 REST1    INC     L               ;04:BUMP VALUE; REST TEST
4F86 2D         00440          DEC     L               ;04:BUMP VALUE; REST TEST
4F87 C28C4F     00450          JP      NZ,REST2        ;10:ONLY 00 DEFINES REST
4F8A CBEF       00460          SET     5,A             ;08:SILENCE VOICE IF REST
4F8C 14         00470 REST2    INC     D               ;04:BUMP VALUE; REST TEST
4F8D 15         00480          DEC     D               ;04:BUMP VALUE; REST TEST
4F8E C2934F     00490          JP      NZ,REST3        ;10:ONLY 00 DEFINES REST
4F91 CBF7       00500          SET     6,A             ;08:SILENCE VOICE IF REST
4F93 1C         00510 REST3    INC     E               ;04:BUMP VALUE; REST TEST
4F94 1D         00520          DEC     E               ;04:BUMP VALUE; REST TEST
4F95 C29A4F     00530          JP      NZ,REST4        ;10:ONLY 00 DEFINES REST
4F98 CBFF       00540          SET     7,A             ;08:SILENCE VOICE IF REST
4F9A 02         00550 REST4    LD      (BC),A          ;07:SET VOICES ON OR OFF
                00560 ;
                00570 ; ******************************************************
                00580 ; DECREMENT H,L,D,E (WAVEFORM DURATION FOR EACH VOICE)...
                00590 ; NEEDED EACH TIME THE WAVEFORM IS TOGGLED DURING LOOPS..
                00600 ; ...INNER LOOP BEGINS HERE.  T-STATES STRICTLY EQUAL 246
                00610 ; MEANING MAXIMUM FREQUENCY IS APPROXIMATELY 1770000/246
                00620 ; OR 7195.1 HZ. USEFUL FREQUENCIES ARE CONSIDERABLY LESS.
                00630 ; ******************************************************
                00640 ;          BEGIN PITCH AND RHYTHM COUNTDOWN LOOPS
                00650 ; ******************************************************
                00660 ;       COUNT DOWN THE PITCH LOOP FOR VOICE NUMBER ONE
                00670 ; ******************************************************
                00680 ;
4F9B 0A         00690 LOOP2    LD      A,(BC)          ;07:WHAT WAVE IS LURKING
4F9C 25         00700          DEC     H               ;04:COUNTDOWN FREQUENCY 1
4F9D C2A04F     00710          JP      NZ,EXIT1        ;10:SAME WAVE IF NOT 0
4FA0 EE01       00720          XOR     1               ;07:TOGGLE WAVEFORM BIT 1
4FA2 DD6602     00730          LD      H,(IX+2)        ;19:RESTORE PITCH VALUE
4FA5 C3AE4F     00740          JP      EXIT1A          ;10:JUMP PAST TIMEWASTERS
4FA8 FDE5       00750 EXIT1    PUSH    IY              ;15:WASTE 15 T-STATES
4FAA FDE1       00760          POP     IY              ;14:WASTE 14 T-STATES
4FAC E6FF       00770          AND     0FFH            ;07:WASTE 7 MORE T-STATES
                00780 ;
                00790 ; ******************************************************
                00800 ;       COUNT DOWN THE PITCH LOOP FOR VOICE NUMBER TWO
                00810 ; ******************************************************
                00820 ;
4FAE 2D         00830 EXIT1A   DEC     L               ;04:COUNTDOWN FREQUENCY 2
4FAF C2BA4F     00840          JP      NZ,EXIT2        ;10:SAME WAVE IF NOT 0
4FB2 EE02       00850          XOR     2               ;07:TOGGLE WAVEFORM BIT 2
4FB4 DD6E03     00860          LD      L,(IX+3)        ;19:RESTORE PITCH VALUE
4FB7 C3C04F     00870          JP      EXIT2A          ;10:JUMP PAST TIMEWASTERS
4FBA FDE5       00880 EXIT2    PUSH    IY              ;15:WASTE 15 BANANAS
4FBC FDE1       00890          POP     IY              ;14:DRUM FINGERS ON 14
4FBE E6FF       00900          AND     0FFH            ;07:USELESS ARITHMETIC
                00910 ;
                00920 ; ******************************************************
                00930 ;       COUNT DOWN THE PITCH LOOP FOR VOICE NUMBER THREE
                00940 ; ******************************************************
                00950 ;
4FC0 15         00960 EXIT2A   DEC     D               ;04:COUNTDOWN FREQUENCY 3
4FC1 C2CC4F     00970          JP      NZ,EXIT3        ;10:SAME WAVE IF NOT 0
4FC4 EE04       00980          XOR     4               ;07:TOGGLE WAVEFORM BIT 3
4FC6 DD5604     00990          LD      D,(IX+4)        ;19:RESTORE PITCH VALUE
4FC9 C3D24F     01000          JP      EXIT3A          ;10:JUMP PAST TIMEWASTERS
4FCC FDE5       01010 EXIT3    PUSH    IY              ;15:SCRATCH LEFT HAND
4FCE FDE1       01020          POP     IY              ;14:SCRATCH RIGHT HAND
4FD0 E6FF       01030          AND     0FFH            ;07:CHECK KITCHEN CLOCK
                01040 ;
                01050 ; ******************************************************
```

Program continues

the frequency range under 200 Hz (cycles per second); trying to use four voices by this method would result in little more than head-pounding sonic thuds.

The answer is, in part, machine language. Look at Program Listing 1; the program begins at 4F60 (decimal 20320). Pitches and rhythms will be stored in a music array beginning at 5000H (20480), so index register IX is set to that value. Since the circuit is mapped to location 4FFF, the BC register is set to that value.

Microprocessor registers are specialized memory locations inside the chip itself. For reasons of speed this program makes use of many of the registers available in the Z-80. To understand why, it's necessary to know how microprocessors work. Certainly they are calculators, but by comparison with the arithmetic powers of chips inside a hand-held scientific calculator, microprocessors are pipsqueaks.

Instead, microprocessors are fast, flexible, general-purpose, switching tools. In response to a combination of binary digits, the thousands of internal gates of the chip will quickly make one of several hundred responses. Simple internal actions can be done quickly; lengthier ones involving reading from, or writing to, memory take more time.

The time it takes a microprocessor to perform any function, then, is dependent on three things: the nature of the instruction, the length of the instruction, and the speed of the computer's master clock. The faster the clock is, the faster the instruction will be completed—at least up to the point at which the circuit components fail to switch on or off fast enough to be reliable.

If we limit the instructions to those that operate on-board the microprocessor chip, we gain speed. But the Z-80 processor is an odd sort of device. It was once described as "an 8080 with wings", because the 8080 had just a few registers and limited vocabulary. If the Z-80 was a true upgrade of the 8080, it would be able to execute all the instructions the 8080 could, and more.

This brings us back to the byte. The byte? What? Sure—because the largest number represented by a byte is 11111111, or decimal 255. That limits the number of one-byte processor instructions, obviously, to 255. In order to be a really nifty upgrade, the Z-80 had to do a lot more than the 8080. So its designers took a few unused instruction bytes (called operation codes, or "op codes"), and used them as pointers to a second instruction byte. Specifically, bytes CB, DD, ED, and FE tell the processor that another byte follows; the combination of the two define a brand

WE HAVE A FULL HOUSE

DON'T GAMBLE

Buy Only From a "Factory Authorized Source"

THE STOCKING SOURCE

PRINTERS	List Price	Your Cost
Okidata Microline 80	$ 800.	Ask for Our Price
NEW Microline 82	$ 960.	Ask for Our Price
Anadex Model DP-8000 or DP-8000AP	$1095.	Ask for Our Price
Anadex Model DP-9500 or DP-9501	$1650.	Ask for Our Price
Epson Model TX-80B Friction Feed	$ 710.	Ask for Our Price
Epson Model TX-80B Tractor Feed & Graftrax	$ 799.	Ask for Our Price
Epson Model MX-80	$ 645.	Ask for Our Price

Ask about the gift that will accompany every printer order placed in December 1980

INTERFACES

- Okidata Microline 80 Tractor Feed . . $100.
- Okidata Microline 80 RS-232 Interface with 256 Character Buffer . . . . . $200.
- All above Printers — Cable from Printer to TRS-80 . . . . . $ 35.
- Epson-Serial Interface & Cable . . . . . $ 75.
- Epson IEEE 488 Interface & Cable Specify Standard or PET Model . . . . $ 80.
- Epson Apple Plug-In Interface with On-Board ROM and Cable . . . . . $110.

ASK FOR OUR INSTANT DISCOUNT
From Roy Hawthorne
Talk To Bill Tokar On Applications

CALL TOLL FREE
U.S.A.
1-800-521-2764
MICHIGAN
1-800-482-8393

WRITE TO: 438
"The Stocking Source"
23995 Freeway Park Dr.
Farmington Hills, MI 48024

```
                    01060 ;     COUNT DOWN THE PITCH LOOP FOR VOICE NUMBER FOUR
                    01070 ; ***************************************************
                    01080 ;
4FD2 1D             01090 EXIT3A  DEC     E               ;04:COUNTDOWN FREQUENCY 4
4FD3 C2DE4F         01100         JP      NZ,EXIT4        ;10:SAME WAVE IF NOT 0
4FD6 EE08           01110         XOR     8               ;07:TOGGLE WAVEFORM BIT 4
4FD8 DD5E05         01120         LD      E,(IX+5)        ;19:RESTORE PITCH VALUE
4FDB C3E44F         01130         JP      EXIT4A          ;10:JUMP PAST TIMEMASTERS
4FDE FDE5           01140 EXIT4   PUSH    IY              ;15:WATER NASTURTIUMS
4FE0 FDE1           01150         POP     IY              ;14:PICK 14 ZUCCHINI
4FE2 E6FF           01160         AND     0FFH            ;07:MIX APPLES AND ORANGE
                    01170 ;
                    01180 ; ***************************************************
                    01190 ; CHECK FOR END OF NOTE DURATION; GET MORE NOTES IF DONE
                    01200 ; ***************************************************
                    01210 ;
4FE4 02             01220 EXIT4A  LD      (BC),A          ;07:OUTPUT NEW WAVEFORMS
4FE5 D9             01230         EXX                     ;04:GET STASHED DURATION
4FE6 0B             01240         DEC     BC              ;06:COUNT DOWN DURATION
4FE7 78             01250         LD      A,B             ;04:SET UP B FOR TEST
4FE8 B1             01260         OR      C               ;04:CHECK AGAINST C
4FE9 D9             01270         EXX                     ;04:STASH DURATION AGAIN
4FEA C29B4F         01280         JP      NZ,LOOP2        ;10:GO BACK TIL NOTE END
                    01290 ;
                    01300 ; ***************************************************
                    01310 ; MOVE ALL POINTERS PAST CURRENT BATCH OF NOTES/DURATIONS
                    01320 ; ...THIS IS THE REMAINDER OF OUTER LOOP, T-STATES = 80,
                    01330 ; TOTAL T-STATES OF OUTER LOOP = 80 + 244 = 324, WHICH IS
                    01340 ; ABOUT 2 OF THE MAXIMUM CYCLE FREQUENCIES (.0002 USEC).
                    01350 ; ***************************************************
                    01360 ;
4FED 110600         01370         LD      DE,6            ;10:MEMORY POS'NS TO MOVE
4FF0 DD19           01380         ADD     IX,DE           ;15:MOVE 6 PLACES FORWARD
                    01390 ;
                    01400 ; ***************************************************
                    01410 ; CHECK FOR END OF PROGRAM CODE (00) OR DEPRESSED BREAK
                    01420 ; ***************************************************
                    01430 ;
4FF2 DD7E00         01440         LD      A,(IX+0)        ;19:NEXT NOTE DURATION
4FF5 B7             01450         OR      A               ;04:SET END-OF-MUSIC FLAG
4FF6 C8             01460         RET     Z               ;05:BACK TO BASIC IF DONE
4FF7 3A4038         01470         LD      A,(3840H)       ;13:TEST BREAK KYBD ROW
4FFA B7             01480         OR      A               ;04:SET FLAG FOR KEY TEST
4FFB CA674F         01490         JP      Z,LOOP1         ;10:CONTINUE PIECE IF OK
4FFE C9             01500         RET                     ;   :TO BASIC IF BREAK
                    01510 ;
                    01520 ; ***************************************************
                    01530 ;
06CC                01540         END     06CCH           ;   :READY AFTER SLASH
00000 TOTAL ERRORS


EXIT1   4FA8 00750   00710
EXIT1A  4FAE 00830   00740
EXIT2   4FBA 00880   00840
EXIT2A  4FC0 00960   00870
EXIT3   4FCC 01010   00970
EXIT3A  4FD2 01090   01000
EXIT4   4FDE 01140   01100
EXIT4A  4FE4 01220   01130
LOOP1   4F67 00230   01490
LOOP2   4F9B 00690   01280
REST1   4F85 00430   00410
REST2   4F8C 00470   00450
REST3   4F93 00510   00490
REST4   4F9A 00550   00530
```

Program Listing 1. Assembly listing for the music performance program to drive the 4FFF sound circuit.

new instruction. This gives the Z-80 upwards of 500 new commands.

The sacrifice, of course, is time. In order to determine the second byte of the instruction, the processor must dutifully "fetch" it from memory.

Fast, on-board Z-80 instructions use the A, B, C, D, E, H and L registers, singly or in pairs. The alternate set of registers (A', B', C', D', E', H', and L') operate at the same speed. The longer instructions involve, unfortunately, the very flexible IX and IY registers.

The IX and IY registers are "index registers." This means that, when we set IX equal to 5000 (as in line 160 of Program Listing 1), we can operate not only at memory address 5000, but within a half byte's distance in either direction. HL + 1, DE + 6, or BC – 28 have no meaning to the microprocessor, but IX + 1 does, and as such it permits more versatile dealings with any block of data.

Let's go back to the listing, at the beginning of the "outer loop." Since we always want BC to identify the circuit port (4FFF), we will exchange registers, saving this information and moving to the alternate set to define B'C'. They take the values stored at IX + 1 (5001), which will become the total duration of a given note. The registers are then swapped back.

The H registers is the pitch value for voice #1, the L register defines voice #2, D is voice #3, and E is voice #4. Each register obtains its value from an array identified by IX + 2 (5002) through IX + 5 (5005).

When discussing the hardware, the purpose of Z2 was in question. Lines 370 and 550 provide the answer. The accumulator retrieves whatever value is stored in BC (i.e., at location 4FFF). In the circuit (Fig. 1), four bits are reserved to turn on or off each of the buffers in Z2. By "masking" the value in A with 0F hex (00001111), the four bits farthest to the right are forced low, and the other pitch bits remain unmolested.

Zero is the value used for a rest. Thus, the combination INC H and DEC H leaves the value in the H register intact, yet setting the Z 80's zero flag.

If the value in H (and later in L, D, and E) is zero, then the appropriate bit in A is set high; if the value in H is not zero, the bit is left alone (remember all the Z2 control bits were set low in line 380). A low bit turns on Z2; a high bit turns it off.

Once the voices have been marked on or off as dictated by the values stored in memory, the byte is written to 4FFF in line 550. Recall that "write to 4FFF" is the hardware signal to action, and the circuit responds by mirroring the value written to memory. The circuit now knows which voices to sound and which to silence. The assigned voice will not change until the next trip through this outer loop.

Facing the Music

Finally the real work begins as the program enters the inner loop. The contents of the BC register (which still points to 4FFF) are retrieved. Four identical routines follow.

There are some important numbers in the comment column (following the semicolons) on each line. These count the number of clock periods (called "T-States") required to execute each instruction. An accurate count of these is critical in music.

Each of the pitch registers is decremented until it reaches zero. In the accumulator, the bit representing that voice is then toggled from its present state to its complement (lines 720, 850, 980 and 1110). If the bit was a one, it is changed to zero, and vice versa. The pitch code is then restored by rereading the note value pointed to by the register IX (lines 730, 860, 990 and 1120).

There is some interesting code that is required before a pitch value reaches zero. Examine lines 750, 760 and 770. These instructions do nothing but waste an amount of time equivalent to the time it

Pensadyne

Bringing Word Processing Power to the People

Performance. At a price that you can afford. The basis on which our company has built a reputation that spans hundreds of software sales in seven countries world wide. We're here to stay, providing the TRS-80 software you want, the service and support you need, and a price tag that a shrinking budget can keep up with.

PENSA-WRITE 2 — A new generation word processing system that conforms to your needs and requirements. It's flexible, versatile, lightning quick, and includes system features unparalled in the industry. Editor features include...

- in memory capacity of over 19,000 characters (approx. 3500 words), automatic chaining of files during printing and all in just 32K of memory!
- full editing capabilities including: global search and replace, fully controlled transparent cursor, selective insert and delete functions, keyword searches, non-printing comments, forward and backward scrolling, visible linefeed characters, complete word wrap around and much, more more...
- user orientation features included on screen. Three screen lines are constantly dedicated to communicating information to the user, or to allnw the user to communicate with the program. These features include: constant read out of date and time, program location, the number of characters that have been used in the text buffer, and the number of characters that remain to be used, automatic word counts, free disk space (on all drives), and a storage indicator that will tell you if what you have in memory has been stored on disk.
- directories for all drives available on screen without exit to DOS
- full error descriptions as well as up to eight possible suggestions as to what might have gone wrong
- sophisticated program structure that will allow the addition of program modules in the future, to further enhance your PENSA-WRITE 2
- complete system and user documentation written in plain English, and supplied in its own three ring binder, so any program updates can easily be added to the documentation.

Your initial purchase of the PENSA-WRITE 2 will include the editor, and at no additional charge, a general purpose printing program. In the future, as they become available and as you require them, you will be able to add options that fully interlock with your editor (with no software modifications). Enhancement modules will include a mailing list, basic file editor, report printer, as well as printing modules for special printers (such as the Centronics 737) and for other special applications. But there's more. Pensadyne Computer Services maintains that after sales services are vital to the full implementation and support of our programs. In the unlikely event that a problem should arise with one of our products, we maintain a 24 hour service department where you can call and get your questions answered. Software support — we guarantee it. **In writing.** We want you to like what we do. Because if you do, then you will come back again in the future. And that's the name of the game.

The price of the PENSA-WRITE 2 word processing system at our special introductory price... **$79.95**

Pensa-write 1 and mailing list, the industry standard BASIC word processor, still available on disk for the low, low price of just **$19.95.**

Mastercharge and Visa Welcome. Order by phone or write.

Dealer Inquiries Invited.

PENSADYNE. Giving you the power to think.

PENSADYNE

COMPUTER SERVICES

4441 WEST FIRST AVE., VANCOUVER, B.C. V6R 4H9 TELEPHONE: 604-224-3107

207

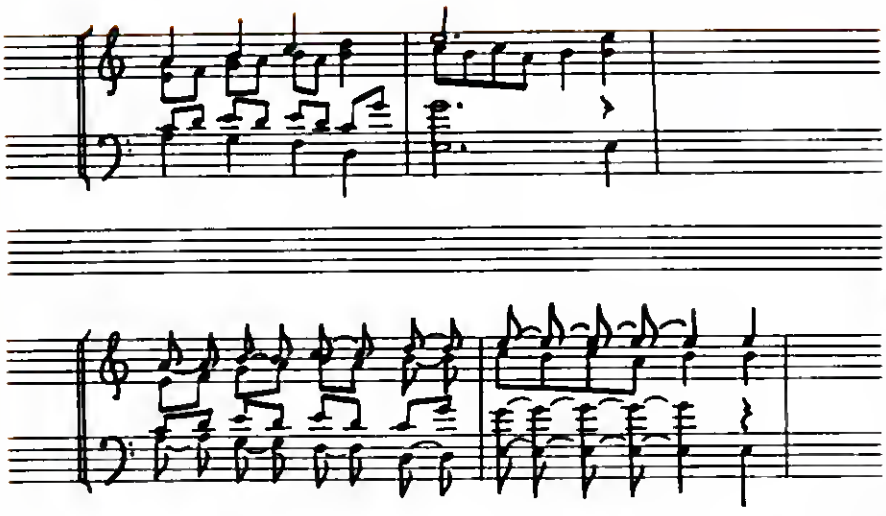


Figure 3. An excerpt from the score translated into data statements in Listing 2, (a) as written in standard music notation, and (b) as transcribed for use with the 4FFF sound circuit.

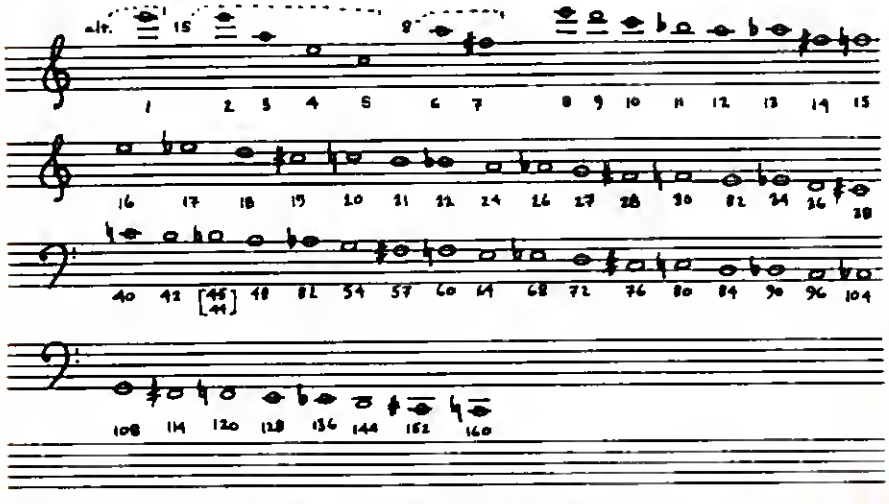


Figure 4. Useful pitches that can be derived by the program in Listing 1. The pitches shown, which are only approximate, are for a TRS-80 with a 50 percent speed up modification installed. Pitches sound a tritone below on an unmodified unit.

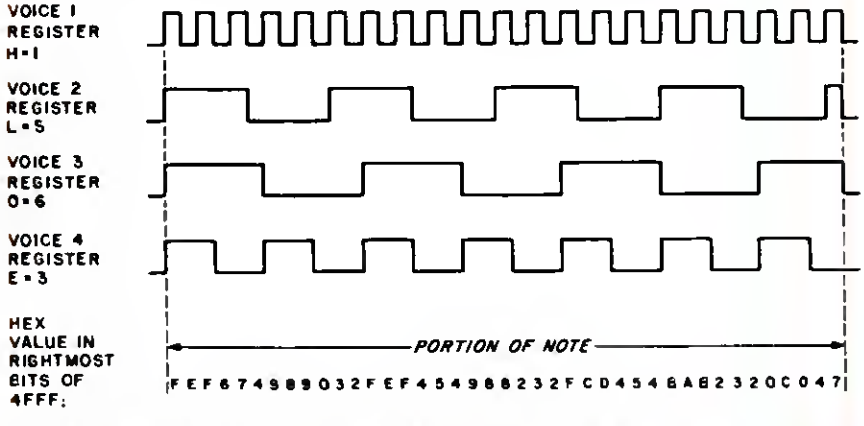


Figure 5. Idealized waveforms present at the output of the circuit, representing the changes to the four high bits in memory location 4FFF.

would take to toggle a bit in the accumulator, restore the original pitch value, and move on. The reason for this is subtle: If we take more time when changing the waveform than when leaving it alone, then a higher frequency (because it toggles more often) will take more aggregate time than a lower frequency. Hence, it will lengthen the loop as a whole, and *lower* the simultaneous pitches in the chord!

When all the testing has been completed for four voices, the result (in line 1220) is written to 4FFF. Thus, whatever waveform differences might have occurred are now transferred to both memory location 4FFF and to the circuit. (See Fig. 5.)

In lines 1230 to 1280, the note duration value is retrieved from the alternate BC register pair, and the loop is repeated until the note is complete. When the note is finished, the index register is moved six places forward to the next block of notes and durations (lines 1370 and 1380).

A duration of zero gives the cue to end the music (lines 1440-1460); if the BREAK key is depressed (lines 1470-1500), the piece also concludes.

The program in Program Listing 2 converts a familiar tune to values which can be read by the assembly language program. Connect the Earie, powerup the TRS-80, set MEMORY SIZE to 20320, and CLOAD the BASIC program. There is just enough room to squeeze it in, but you must CLEAR0 before running it. Its only job is to read the pitch values for the individual voices and durations and POKE them in place starting at 20480 (5000 hex). The starting address is also put in place.

Load the SYSTEM music subroutines,

Integer	Result	Integer	Result
160	44 970	40	179 878
152	47.336	38	189 345
144	49 966	36	199 864
136	52 905	34	211.621
128	56.212	32	224 848
114	63 115	28	256.969
108	66.622	27	266 486
104	69 184	26	276.735
96	74 949	24	299.797
90	79 946	22	327 051
84	85 656	21	342.625
80	89.939	20	359.756
76	94 673	19	378.691
72	99 932	18	399.728
68	105.811	17	423.242
64	112.424	16	449 695
60	119 919	15	479.675
57	126.230	14	513 937
54	133 243	13	553.471
52	138.368	12	599 593
48	149 896	11	654.102
44	163 525	10	719 512
42	171.312	9	799 458

Table 2

```
100 Q=28480
110 FOR X = Q+2 TO 22000 STEP6
120 READ A : IF A=255 THEN 380
130 POKE X,A : NEXT
140 DATA 0,0,0
150 DATA0,0,0,0,0,0,0,0,0
160 DATA0,0,0,0,0,0,24,0
170 DATA24,0,16,0,16,0,16,0
180 DATA20,0,21,0,24,0,27,0
190 DATA24,0,21,0,20,0,18,0
200 DATA16,16,16,16,16,24,0
210 DATA24,24,16,0,16,16,18,0
220 DATA20,0,21,0,24,0,27
230 DATA24,24,21,21,20,20,18,18
240 DATA16,16,16,16,16,16
250 DATA15,18,16,15
260 DATA13,12,16,16,18
270 DATA20,24,21,20
280 DATA18,18,20,18
290 DATA16,16,16,16,15,15,16,16
300 DATA16,16,18,18,20,20,21,21
310 DATA24,24,24,24,20,21,24,0
320 DATA18,18,18,20,18
330 DATA16,16,15,15,13,13,12,12
340 DATA16,18,20,21
350 DATA24,24,24,24,24,24
360    DATA 24,24,24,24,24
370 DATA24,24,24,0,255
380 REM * VOICE B
390 FOR X = Q+3 TO 22000 STEP6
400 READ B : IF B=255 THEN 660
410 POKE X,B : NEXT
420 DATA 0,0,0
430 DATA0,0,0,0,32,0,0,0
440 DATA0,0,0,0,32,0,0,0
450 DATA32,0,0,0,24,0,0,0
460 DATA32,0,0,0,24,0,0,0
470 DATA32,0,0,0,24,0,0,0
480 DATA20,21,20,24,21,24,0
490 DATA32,32,0,0,24,24,0,0
500 DATA32,0,0,0,24,0,0
510 DATA32,30,27,24,21,24,21,21
520 DATA20,21,20,24,21,21
530 DATA24,0,27,0
540 DATA32,0,40,20,21
550 DATA24,30,27,27
560 DATA28,28,27,30
570 DATA32,32,32,32,24,0,24,0
580 DATA20,20,21,21,24,24,26,26
590 DATA32,32,32,32,32,0,32,32
600 DATA24,24,24,0,0
610 DATA32,32,32,36,27,27,27,28
620 DATA26,24,28,26
630 DATA24,27,30,32,30,36
640   DATA32,32,32,32,32
650 DATA32,32,32,255
660 REM * VOICE C
670 FOR X = Q+4 TO 22000 STEP6
680 READ C : IF C=255 THEN 940
690 POKE X,C : NEXT
700 DATA0,0,0
710 DATA0,0,0,0,40,0,0,0,0
720 DATA0,0,0,0,48,0,0,0
730 DATA40,36,32,0,40,0,0,0
740 DATA40,36,32,0,40,0,0,0
750 DATA40,36,32,0,40,0,0,0
760 DATA27,27,27,27,27,0,0
770 DATA40,36,32,0,40,0,0,0
780 DATA40,36,32,0,40,0,0
790 DATA40,36,32,36,32,36,40,30
800 DATA27,27,27,27,27,0
810 DATA36,36,54,54
820 DATA42,42,64,64,64
830 DATA48,48,42,30
840 DATA36,36,32,42
850 DATA40,40,40,40,40,40,0,0
860 DATA40,36,32,0,40,0,0,0
870 DATA40,36,32,0,40,36,40,42
880 DATA48,48,48,0,0
890 DATA40,36,32,0,40,36,40,0
900 DATA42,42,42,42
910 DATA40,40,40,40,40,40
920   DATA40,40,40,40,40
930 DATA48,48,48,255
940 REM * VOICE D
950 FOR X = Q+5 TO 22000 STEP6
960 READ D : IF D=255 THEN 1220
970 POKE X,D : NEXT
980 DATA64,60,54
990 DATA48,0,64,0,64,0,64,60,54
1000 DATA48,0,64,0,64,0,64,0
1010 DATA96,0,64,0,64,0,60,0
1020 DATA96,0,64,0,64,0,84,0
1030 DATA96,0,64,0,64,0,84,0
1040 DATA64,64,64,64,64,64,0
1050 DATA96,0,64,0,64,0,60,0
1060 DATA96,0,64,0,64,0,84
1070 DATA48,48,54,54,60,60,72,72
1080 DATA64,64,64,64,64,64
1090 DATA72,72,88,80
1100 DATA84,84,96,96,96
1110 DATA96,96,108,96
1120 DATA144,72,54,54
1130 DATA80,0,54,0,54,0,80,0
1140 DATA96,0,60,0,64,0,128,0
1150 DATA96,96,96,0,80,80,80,0
1160 DATA60,64,72,80,84
1170 DATA96,96,60,64,60,64,72,72
1180 DATA64,64,128,128
1190 DATA96,96,96,96,96,96,96
1200   DATA0,64,60,54
1210 DATA48,64,96,255
1220 REM * RHYTHMS
1230 FOR X = Q TO 22000 STEP6
1240 READ E : IF E=255 THEN 1490
1250 POKE X,E:POKE X+1,100 : NEXT
1260 DATA3,3,3
1270 DATA4,4,4,4,4,4,3,3,3
1280 DATA4,4,4,4,4,4,4,4
1290 DATA4,4,4,4,4,4,4,4
1300 DATA4,4,4,4,4,4,4,4
1310 DATA4,4,4,4,4,4,4,4
1320 DATA4,4,4,4,8,4,4
1330 DATA4,4,4,4,4,4,4,4
1340 DATA4,4,4,4,4,4,8
1350 DATA4,4,4,4,4,4,4,4
1360 DATA4,4,4,4,8,8
1370 DATA8,8,8,8
1380 DATA8,8,4,4,8
1390 DATA8,8,8,8
1400 DATA8,8,8,8
1410 DATA4,4,4,4,4,4,4,4
1420 DATA4,4,4,4,4,4,4,4
1430 DATA4,4,4,4,4,4,4,4
1440 DATA4,4,8,8,8
1450 DATA4,4,4,4,4,4,4,4
1460 DATA8,8,8,8
1470 DATA3,3,3,3,5,5,5,6,7,7,7
1480 DATA11,14,42,0,0,255
1490 POKE16526,96:POKE16527,79
```

Program Listing 2. BASIC listing of a familiar holiday tune to be used in conjunction with the machine-language driver in Listing 1.

and either BREAK or enter a slash ("/"). The piece is ready to play. Connect the circuit's cables to a high-fidelity audio amplifier, and type:

```
PRINT USR(0)
```

Well, it seems lively enough, but why are the pitches so low? Look at Table 2. The maximum frequency that an unmodified TRS-80 can produce using this program is 719 5.12 Hz, which means the program loops through its actions more than 7000 times per second. By itself, this is a very high frequency, nearly double the highest playable note on an acoustic instrument.

The difficulty arises when we are forced to use one of 255 possible values through which to send our pitch loop. This means that the only possible pitch values are 7195.12 divided by one through 7195.12 divided by 255. The smaller number of divisions aren't close to a traditional scale, although the notes are high. The larger numbers yield pitches that are fairly in-tune, but also quite low.

You might be forced to think of the melody as being sung by a group of very raspy baritones. Another option is a hardware speed-up to the TRS-80 (see *80 Microcomputing*, Feb., 1980). This will raise the pitch a half octave. Another option is to use a retriggerable flip-flop at the far end of Z2. This requires one more integrated circuit to reshape the waveforms and make them audible.

There are also a few software methods, but they reduce the attractiveness of the program. The extraction of a voice will raise the pitch; the extraction of two voices will raise it further. At last, a single voice can be produced which will open up a great portion of the traditional scale. Just think—if another three TRS-80's turn up for the holidays...

It is possible to create a look-up table by compiling the score as it is input, before it is performed. In that way, a composite monaural sound can be produced that is relatively in tune and higher in pitch. However, this method is sophisticated and certainly outside the scope of "Applications."

Of course, the realm of quadriphonic, three-dimensional audio sound effects is still available, and perhaps this is the best use of the Earle.

If you plan to use the circuit with an audio mixer, POKE 20479 (4FFF) with zero before starting; this will get all the voices in phase (i.e., starting at the same time). Likewise, you can experiment with phasing by altering the value in 4FFF before beginning a piece, of at the start of each note.

Those with an expansion interface can save the trouble of building the hardware at the cost of lowering the pitches still further. The pitch value can be stored at 4FFF, but also loaded into 37E8, which is already mapped to the printer port address. Just hook up some resistors to the edge card, and it's ready to go. Of course, you can't use rests in this configuration.

Creating Your Own Tunes

Putting together your own music is time-consuming but straightforward. Fig. 3 is an excerpt from my arrangement of the tune in Program Listing 2. These measures are written two ways: One is standard musical notation, and the other is notated to use with the hardware.

Since the machine language program uses a single loop to produce all four voices, it follows that the loop is concluded at the termination of the shortest note in a harmonic group. That's why a note must be redrawn on the score—as a reminder to include in in the next loop. Also,

each rest is counted as a separate "note of silence," requiring that each rest be no longer than the shortest note played simultaneously with it.

Once you are certain that you've got the score broken into four lines of equal parts, you can create the BASIC POKE program. The pattern is six bytes long, two for duration, and four voices. The note's duration takes two bytes. The first byte must be at least 1, and the second can be any value. I maintain the second byte at 100 (64 hex) in this example, but it can be used to shade the rhythm with rubato.

Finally, the voices follow in order, one byte for each pitch or rest in the harmonic structure. A zero in any voice position defines a rest, and a zero in the first duration position defines the double bar.

Good luck, and here's hoping you like those raspy baritones!

Personal Thoughts

This month's Applications completes my first year with *80 Microcomputing*. During that time I have been rewarded with hundreds of letters and telephone calls from readers with suggestions and questions. Since early spring, every column has been based on suggestions from readers, and there are many more yet to address. So during 1981, expect to discover how to add ROM and RAM to your TRS-80; a step-by-step on converting a machine language program to BASIC POKEs and strings; what to do when your system stops working; high-resolution graphics (What did he say????); single-keystroke subroutines; and replacing your BASIC ROM with a monitor of your own making. I look forward to hearing from you, and wish all you remarkable, diverse TRS-80 users the very best during this season and the coming year.■

```
10 FOR X = 20320 TO 20478 : READ A : POKE X,A : NEXT
20 DATA 221,33,0,60,1,255,79,217,221,78,0,221
30 DATA 78,1,217,221,102,2,221,110,3,221,86,4
50 DATA 221,94,5,10,230,15,36,37,194,133,79,203
60 DATA 231,44,45,194,140,79,203,239,20,21,194,147
70 DATA 79,203,247,28,29,194,154,79,203,255,2,10
80 DATA 37,194,160,79,238,1,221,102,2,195,174,79
90 DATA 253,229,253,225,230,255,45,194,186,79,238,2
100 DATA 221,110,3,195,192,79,253,229,253,225,230,255
110 DATA 21,194,204,79,238,4,221,86,4,195,210,79
120 DATA 253,229,253,225,230,255,29,194,222,79,238,0
130 DATA 221,94,5,195,228,79,253,229,253,225,230,255
140 DATA 2,217,11,120,177,217,194,155,79,17,6,0
150 DATA 221,25,221,126,9,183,200,58,64,56,183,202
160 DATA 183,79,201
```

Program Listing 3. BASIC listing that will POKE in place the assembly language driver for the 4FFF sound circuit. Once this program has been run, it may be deleted to make room for Listing 2.

80 NEWS

edited by Nancy Robertson

Model I Caught By FCC Fallout

Have you heard that the Model I is being discontinued? That's the latest piece of gossip traveling the industry grapevines.

There is reason to speculate on an early death for the first born of Tandy's computer line: Model I micros are in short supply, and the Model I must be remodeled to meet Federal Communication Commission (FCC) regulations that will go into effect Jan. 1, 1981.

Will the Model I be discontinued? According to Tandy/Radio Shack's John Shirley, who heads the computer division, "That's not the sort of thing we'd like to comment on one way or the other." Ed Juge, another Tandy exec, did mention "the difficulty of guaranteeing shipment after the first of the year."

Shirley claims that Model I computers are presently "in short supply because we are later than we anticipated with the Model III, and it has put a strain on production." But FCC compliance is still up in the air.

The FCC first considered regulating low power communicating devices for radio frequency interference (RFI) in 1976. After three years of study, rules limiting radio frequency (RF) emissions were adopted as amendments to Part 15 of the Chapter 47 laws (laws under the jurisdiction of the FCC). The amendments, rather than covering the broad range of electronic communicating devices, regulate computer RF emissions only. They are particularly strict for personal computers.

The problem originates with conflicting uses of the electromagnetic spectrum, which carries television and radio signals. The extremely quick electronic signals and pulses that are the basis of computer operations create high frequency radio waves. Circuits and traces sometimes act as antennae for these waves. Unless filtered, computer generated RF interferes with radio and TV transmission.

In delineating the need for regulation, the FCC has divided computers into two broad categories: Class A and Class B. A Class A "computing device is marketed for use in commercial, industrial or business environment(s)"; and a Class B "computer device is marketed for use in a residential environment notwithstanding use in commercial, business and industrial environment(s)."

The term "computer device" is meant to stretch to the realm of peripherals, which are also required to comply by Jan. 1 if they are marketed for consumers.

A study of the Part 15 amendments, conducted by Wewer & Mahn for the Micro Industry Trade Assoc., points out that the "dual classification scheme is rooted in the theory that Class B (consumer) devices are in closer proximity to radio, TV, and in many cases, land mobile services than Class A (commercial) devices and thus have a higher potential for causing interference."

Restrictions on Personal Computers

For this reason and others, the FCC is imposing heavier restrictions on home computers. Gene Smerle, the technical editor of *73*, a magazine for ham radio enthusiasts, estimates that Class B radiation limits are over 3 times more demanding than Class A limits. His calculations are based on the figures given in Table 1, which are taken from sections .810 and .830 of Part 15. Wewer & Mahn have interpreted Class B limitations to rule out RFI from 450Kh to 1000Mh—practically the whole broadcasting range.

Actual compliance to the new rules is also less stringent for Class A products in the view of the Wewer & Mahn law firm. Manufacturers are required to "verify" Class A compliance—a lax measure in comparison to "certification" which is required for class B devices. Certification is granted on the basis of testing, application forms and fees. Verification is granted on the basis of the manufacturer's word about test results.

The Effect on Prices

Depending on the amount of research and development required to meet the regulations, it is likely that FCC compliance will result in price hikes for several products.

John Shirley, the corporate head of the computer division of Radio Shack, says that, "There's no question a substantial amount of money has been spent on R and D because of this." Shirley believes the price of some products will remain unchanged, but that other prices are bound to reflect the added work.

Price increases will not affect the new color computer, which has been registered with the FCC as a TV interface device under separate FCC guidelines. Although cost has been added to the manufacture of the Model III, Shirley does not believe it will warrant a price hike for customers. "The Model II is a Class A device in our opinion," Shirley says, and should

Continues to page 56

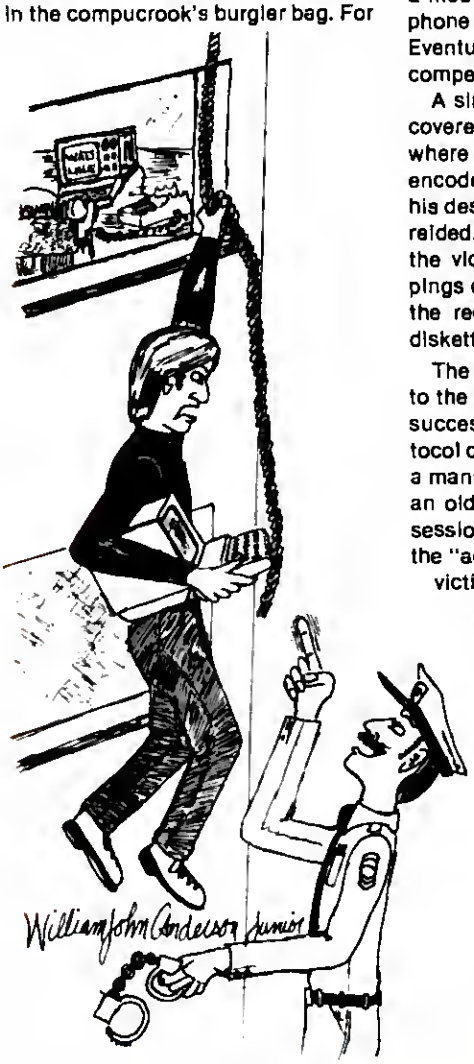
Micros Spotted in Crime Lineup

That crime has kept pace with technology is an inescapable fact of life, but the problem has assumed a whole new dimension with the advent of computer science. The nub of the problem is that business and government have all too frequently plunged into the computerization of their operations with little or no regard for security.

It is not a small problem nor is it a simple one. Estimates of losses due to computer crime run anywhere from $100 million to $40 billion annually, but the rather disturbing truth of the matter is that no one really knows how much is being siphoned off.

Microcomputer Crime

Heretofore, computer crime has been primarily limited to the realm of mainframes and minicomputers, but microcomputers are fast becoming a favored tool in the compucrook's burglar bag. For example, a micro might be programmed to mimic a terminal and thereby secure access to a sensitve data bank. It may also be used to clandestinely duplicate a sign-on routine just long enough to obtain the large computer's password. The thief may then interrogate perhaps thousands of systems at his convenience.



Microcomputer-related crime may also take the form of an automated "cottage industry." Such was the case recently in Pennsylvania where John "Cap'n Crunch" Draper wes arrested for "phone freaking," spoofing Ma Bell's dial codes to make free use of the phone lines.

Although the offense is not new, Draper's updated version of it was more sophisticated than previous methods in that he used a microcomputer. Utilizing a highly involved program, Draper interfaced his Apple II with his home phone via a modem. He was then eble to scan the phone system for operating WATS lines. Eventually he wes detected by the phone company's monitoring equipment.

A simpler form of micro crime wes uncovered earlier this yeer in Tulsa, OK where a bookie had neatly and efficiently encoded ell of his illicit transactions on his desk-top computer. His operation was reided, but, much to the consternetion of the vice squed, none of the usual trappings of a bookie joint were apparent. All the records were maintained on a few diskettes.

The police lugged the equipment back to the station house where they tried, unsuccessfully, to crack the computer's protocol code. Falling in this they summoned a manufacturer's rep. In a 1980 version of an old "bright lights and rubber hoses" session the rep successfully interrogated the "accomplice," paving the way to conviction of the bookie.

> All of this is Greek to the public at large and, predictably, the person on the street tends to be skeptical of that which he does not understand. Recently, a survey entitled "Dimensions of Privacy" was performed by Weston Assoc. for Century Insurance, Inc. Among the results are these three items which serve to illustrate the somewhat uncomfortable feelings many people have about computers in general.
>
> 54 percent of the respondents now believe that computers are a threat to privacy; 63 percent feel that the use of computers should be sharply curtailed to preserve privacy; and 51 percent state that in 10 years people will have lost much of their ability to keep their lives private.

Laws on Computer Abuse

State and federal lawmakers have introduced several proposals designed to curb computer abuse. Sen. Abraham Ribicoff (D-Conn) has sponsored S-240, The Federal Computer Systems Protection Act, now being studied by the Senate Judiciary Committee. Originally introduced in 1976, the bill has since undergone substential rewording to more precisely deal with the technicalities of the ereas it covers.

The bill in its present form was drafted by Philip R. Manuel, an investigative consultent in the field of white collar crime and for 11 years chief investigator for the U.S. Senate's Permanent Subcommittee on investigation.

"Computer crimes (controls) have to date been shoehorned into existing by inadequate laws dealing with crimes ranging from mail fraud to obscene phone calls," said Manuel. "But this bill clearly defines computer crime as computer crime and affords a large measure of protection to the computer systems of the federal government, financial institutions, and all businesses which conduct interstate commerce. It further envisions protection for sophisticated electronic funds transfer systems whose vulnerability to computer fraud is enormous."

Several laws have been enacted on the state level to define and control computer crime. California's legislation in this area is generally acknowledged to be among the best to date, although it is not above criticism.

Don Parker, a computer crime expert with SRI International in Menlo Park, CA finds himself in agreement with most of the California's computer crime statute but, takes exception to the law's definition of computers. He says it excludes "progremmable pocket calculators with atteched external memory devices." In Parker's opinion, the clause is vague and it constitutes a week spot in an otherwise good piece of legislation.

Combatting Compucrime

Effective methods of combatting compucrime ere, of course, available. Maintaining the physical security of the computer can be divided into four broad classifications: 1) controlling the entrance to

Continues to page 56

Competency Tests Processed by TRS-80s

At Conant High School, Jaffrey, NH, a TRS-80 is being used to process and record state mandated tests. Administrators et Conant believe it is the first 80 applied to Competency Testing. The tests are required in New Hampshire, as they are in 38 other states.

The tests are mandated under NH State Statute 186.5, Section Six, which delineates the powers and duties of school boards. The law has been on the books since 1973, however, it was only in 1977 that the state implemented a set of guidelines called the Accountability Plan under which competency testing falls.

The Accountability Plan has six steps with which each school district in the state must comply. They include performance indicators, assessment, analysis of data (Specifically, the state wants to know what the proficiency level is for a whole school in math, language arts, history and science.), and a management plan which outlines district plans for improvements in levels of proficiency throughout the year.

Do these tests contribute to the level of proficiency? Since Competency Testing was first implemented about four years ago, the middle-school students in the Jaffrey-Rindge district have steadily increased their performance level on the Stanford Achievement Test, a standardized norm test given nation-wide. The overall average fell into the ninth stanine in 1979, which is in the highest percentile of echievement, Larry Bramblett, director of instruction for the district, said that when competency testing first started students placed in about the fifth stanine, which is in the average range.

Putting 80s to Work

This particuler school district is ahead of its time—not only in instituting the test but also in their method of compiling results and making them as timely as possible.

John Davys, senior consultant to the NH State Office of Education, and the administrators of this school district feel that the micro is vital to how successful these specialized tests can be.

Davys feels that the Conant project is significant on a statewide level because, "It is unique in the sense that they have maximized the use of technology. Schools have one of the largest stores of human resource and this resource bank shouldn't be tied up with mundane tasks. Technology does those tasks more accurately and quickly and allows the staff to work with the students—which is the way it should be."

At a meeting of the Joint Management Council of educators from all over the stete, Keith Burke, chairman of the council and principal of Conant H.S., gave a demonstration of how their TRS-80 has helped the district menage Competency Test results.

"The council is there," said Burke, "to help other districts implement Competency Tests. They (the council) are always looking for a better way to do it and one way is through managing and keeping track of the data." That's where the 80 comes in.

"The most important thing is keeping accurate records of the testing results or else the whole system goes out the window. The computer handles this very well," said Burke. "Also, it's within 98 percent of the school districts' budgets."

Bramblett said that compiling the results without the computer took too much time and it was also costing the district about $7,000 annually to have someone work on them full time. The whole system—Radio Shack Line Printer III, TRS-80 48K Level II—including the program, cost the Jaffrey-Rindge district roughly $5000.

In New Hampshire, Competencies are taken in grades three, seven and 10 and are given et least three times a year. The way the system works, if a student passes all four areas of testing, for example, in the seventh grade, that student will not have to take the exams again until the 10th grade. If a student doesn't pass the Competency in a particular area, science for example, but passes in the other three areas, that student will only retake the failed test until it's passed.

How Conant Discovered the 80

Conant H.S. has owned a 8K PET, for the past few years. Both Burke and Bramblett saw what the most basic of micros could do and decided that a computer was what they needed in order to compile and turn around test results quickly.

After checking out both APPLE end PET systems, they were told by both dealers that it would be difficult converting the program. Because the program was originelly written on a TRS-80 by programmer Peter Wells, and because he highly recommended the TRS-80, it became the logical choice.

Dave Bramblett Praising Micros to Educators

Some of the specifics of the program are updating, displaying and adding to student records of the Competency Test results. A summary record for the whole school, a class or en individual can be displayed and printed. All of these categories are represented by percentage rates. The program also generates a mailing list to parents.

One disk contains the test results of all the students in a particular school. The information on the data disks are safeguarded by program disks which are protected by code words. The disks are duplicated and put into a safe.

Robert L. Brunelle, commissioner for the New Hampshire State Department of Education, feels that the use of microcomputers to process testing results can be a valuable tool.

"The system is ell interconnected," says Brunelle. "Each district must report to the commission. From this data, the commission does a statewide sampling and from the sampling reports to the Legislature, and in turn the Legislature acts accordingly with the overall findings."

The use of micros in the school districts to compile test results could expedite what can normally be a long and tedious bureaucratic process.

Besides taking care of the paperwork of Competency Tests, Burke is working on programs that will do scheduling and report cards.

Some of the students at Conant H.S. have also become fascinated with the many uses of the TRS-80. "For example," said Burke, "the student council president was having trouble keeping track of the inventory in the school store, so he wrote a progrem to take care of the problem. It (the TRS-80) is a fantastic teaching tool."■

By Pamela Petrakos
80 Staff

Readers' Digest Swallows the Source

In a jointly issued press releeee spokesmen for the Reader's Digest Assoc., Pleasantville, NY and the Source Telecomputing Corp., McLean, VA, announced the Source's acquisition by the Digest for an undisclosed amount. Terms of the acquisition were not made public and spokesmen for both organizations are extremely reticent when queried about the deal.

Rumors of the Source's financiel woes have been rife for several months, and if the microcomputing grepevine is to be believed, the reasons for the Source's take over are likely to lie in its own financial problems.

The event is newsworthy in light of parties involved: The Digest is a multimillion dollar publishing conglomerate, and the Source is a pioneer in microcomputer network technology.

In a carefully worded press release an unidentified spokesman for the Digest is quoted as follows: "The service which can be rendered in helping to expand the delivery of education, health care services, information and knowledge via cable systems, telephones, satellites, etc., is thoroughly consistant with our publishing philosophy."

Several words stand out: Education, health care services, cable systems, satellites. It appears that someone within the Digest organization has big plans for the Source. What these plans are will remain conjecture until both organizations decide to lift the veil of silence they have painstakingly maintained. Everyone who is anyone within the Digest organization prefers not to comment. Spokesmen for the Source have proved equally taciturn, and one can only wonder why.

Logical Merger

The merger of a publishing conglomerate and a computer network is quite logical. This type of arrangement reflects current trends within the publishing industry regarding small company acquisition by larger organizations and efforts by large corporations to diversify their operations as a hedge against the declining economy.

Jack Taub, chairman of Source Telecomputing Corp., says in the joint press release, "We could not have found a better partner than the *Reader's Digest*." He is probably correct. The vast financial resources the Digest has at its disposal and the business acumen it brings to the computer network industry are formidable. The impact this merger will have on the Source's 7000 present customers is unclear, however.

One thing is obvious. Changes are taking place within the computer network industry.

Though a clear picture of what can be expected as a result of the Source's takeover has yet to develop, the doings in Pleasantville and McLean indicate one thing—this might be a good year to ask for a modem for Christmas. ■

by Chris Brown
80 Staff

Campaign Applications: Did Computers Influence Voters?

What really went into the Presidential campaign? Did we choose a winner for the intelligence, integrity and capability of the candidate, or did we judge the product of a computer inspired version of the perfect politicien?

John Cragan and Donald Shields, professors of communications at Illinois State University and the University of Missouri, developed a computer program that analyzes demographic statistics and opinions polled from a given geographical area. The program then chooses among several versions of statements addressing current political situations and arranges e campaign speech that should appeal strongly to the average voter in the polled area.

In the September 22 issue of *Computerworld*, Cragan is quoted as saying, "I'm sure that almost every [candidate] out there today is using a variation of this. I don't think it's as sophisticated or as cynical, but it's something that is used to pretest [statements and ideas] before you have a candidate saying them."

Before the nation made its choice at the polls, I spoke to campaign workers at the national headquarters of the three major presidential candidates. Each campaign made use of computers in several applications; none admitted to using them to the extent suggested by Cregan, however.

Carter's Camp

The Carter campaign probably had the most organized and effective applications. Bill Krause was the Director of Information Services at the national headquarters. He had a staff of three, himself, one of Carter's sons, and a 19-year-old who came on the staff and was trained in BASIC.

In-house, the campaign used Tektronix microcomputers for standard data processing. With these, they kept files on all personnel and volunteers: skills, when they were available for work, etc.

The largest day to day job tackled with the computers was scheduling. The schedule of who should be where doing what changed often—particularly in the last few weeks of the campaign. Members of the Carter family on the campaign trail traveled with a terminal, and checked scheduling changes daily through the campaign's mailbox at The Source.

The Carter campaign also used the New York Times Info Bank, with which they've had a contract since 1969. The Info Bank was used to do research on the other candidates, and to scan news stories for keywords concerning Reagan, military force, etc. The campaign people received abstracts of articles containing pertinent keywords, and these facts were used in turn for campaign speech writing. This method of research and fact gathering greatly reduced the work involved in tracking Carter's opponents, and dropped the necessary information into the laps of Carter's speech writers.

Krause said the general election budget was done on the G. E. time sharing system. IBM System Six word processors were used for personalized letters and other mail.

The Anderson Campaign

John Boswell, EDP Coordinator (among other things) for the Anderson campaign, described three computer applications used in that office. Ninety-five percent of the budget was used to keep computer files on contributers and supporters of the Anderson campaign: contribution history, general personal characteristics, income, and other statistics which could be used by state campaign offices looking for local volunteers and canvassers.

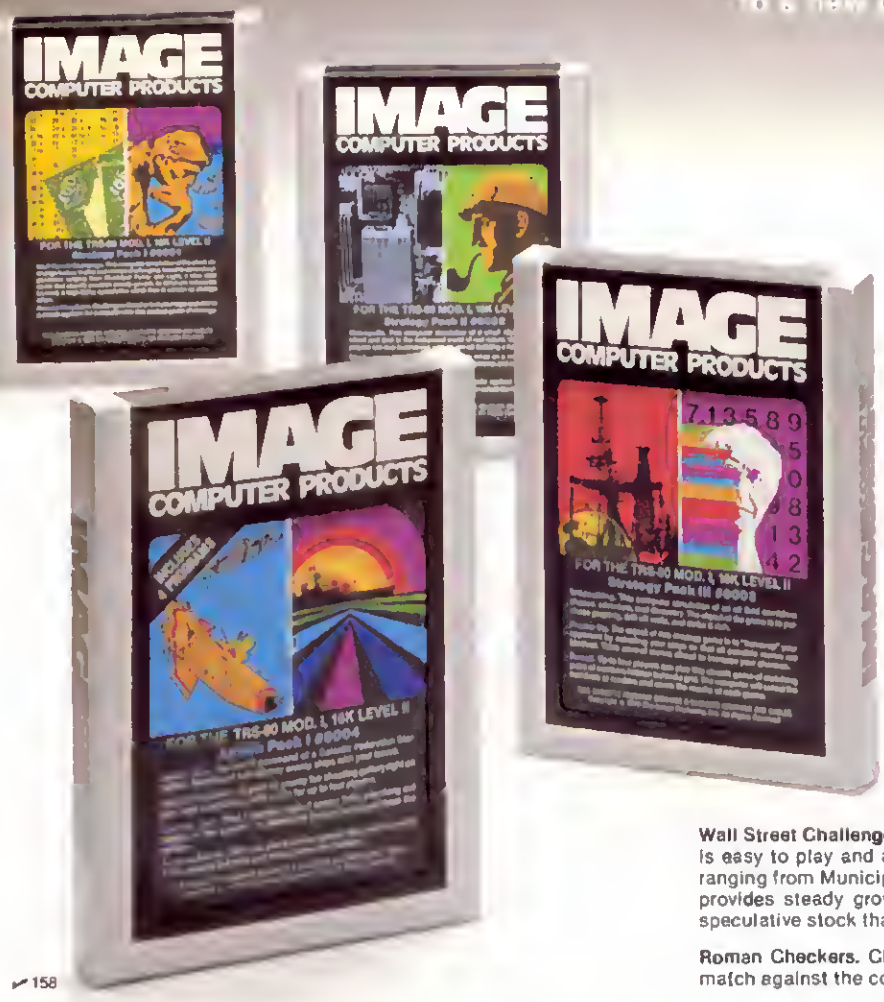
Other applications mentioned by Boswell were payroll and disbursement records. All of these applications were in turn used again to prepare internal management reports and the required income reports to the FCC.

The Anderson campaign was under

Continues to page 56

Put an IMAGE™ on your TRS-80

These cassette programs will introduce you to a new generation of quality software for your 16K Level II TRS-80



Everything from Fast-action animated skill games through mind boggling Strategy and Simulation programs is included in this software collector's series.

Each package contains a quality program cassette in a protective storage box, and complete operating instructions.

These programs run on a 16K Level II TRS-80 Model I.

Strategy Pack I #8001

Wall Street Challenge. This computer simulation of the stock exchange is easy to play and always challenging. Invest in several corporations ranging from Municipal Power and Light, a blue chip stock that usually provides steady growth, to Offshore Industries Limited, a high-flying speculative stock that is certain to change often.

Roman Checkers. Challenge a friend or test your logic and skill in a match against the computer with this ancient game of strategy.

Strategy Pack II #8002

Metropolis. This computer simulation of a small city lets you wheel and deal in the fast-paced world of real estate. Up to eight players can buy businesses with an eye on building a fortune.

Mindmester. This classic strategy game takes on a new dimension as the computer designs the hidden problems and reports the results of each guess.

Wordmaster. Multiple players may compete against the computer to find the hidden word. Each player can select the level of difficulty that matches his individual skill.

Strategy Pack III #8003

Wildcatting. This computer simulation of an oil field combines chance, adventure, and discovery. The object of the game is to purchase property, drill oil wells, and strike it rich.

Frame Up. The object of this strategy game is to "frame-up" your opponent by selecting your plays so that all possible moves are blocked. Think several moves ahead to increase your chances of winning.

Recall. Up to four players can play this classic game of matching pairs of numbers hidden behind a grid. The computer will select the numbers at random and score the results of each guess.

Action Pack I #8004

Space Ace. You are in command of a Galactic Federation Starfighter. Search out and destroy enemy ships with your lasers.

Shooting Gallery. A good ol' county fair shooting gallery right on your own computer. Loads of fun for up to four players.

Bomber Run. Pilot a bomber behind enemy lines searching out targets on the ground. Or, defend the ground and shoot down the bomber.

Air-Sea Battle. Pilot your plane over an enemy ship and try to sink it. Or, captain the ship and shoot down the bomber.

MAKE YOUR CHECK OR MONEY ORDER PAYABLE TO:
Imaga Computer Products, Inc.
615 Academy Drive
Northbrook, IL 60062
*NO C.O.D.'S

PLEASE PRINT
NAME _____
ADDRESS _____
CITY_____ STATE_____ ZIP_____
SIGNATURE _____
PLEASE SEND _____ PACKAGES INDICATED

			QTY:	PRICE
8001	STRATEGY PACK I	$19.95 ea.		
8002	STRATEGY PACK II	$19.95 ea.		
8003	STRATEGY PACK III	$19.95 ea.		
8004	ACTION PACK I	$19.95 ea.		
	ANY COMBINATION OF TWO	$36.95		
	ALL FOUR	$69.95		
		IL RES. AD 6% TAX		
		SHIPPING & HANDLING		$1.00
		TOTAL		

VISA ☐ MASTER CARD ☐
MONEY ORDER ☐ CHECK ☐
CARD # _____
EXPIRATION DATE _____

158

FCC-Model I Dance

Continued from page 51

not be affected by the Class B deadline. Table 2 charts the current Part 15 status of Radio Shack computers.

Shirley made no mention of the Model I in reference to expected price changes. It is questionable whether or not the time and money required to modify the Model I warrant compliance. How much is a hobbyist willing to pay? And what about the Model III option? These are bound to be the thoughts of Tandy's top brass.

Interim Labels

Dave Garner, Tandy's liason with the FCC, says, "We will not make a computer that does not meet compliance after January first." In his opinion the company has three options :1) interim labels; 2) redeveloping products; and 3) dropping products. Products that are marketed after Jan. 1, 1981 that do not meet Class B specifications must carry the following label permanently attached:

This equipment has not been tested to show compliance with new FCC Rules (47 CFR Part 15) designed to limit interference to radio and TV reception. Operation of this equipment in a residential area is likely to cause unacceptable interference to radio communication requiring the operator to take whatever steps are necessary to correct the interference.

Garner explains that there is currently no time limit on the use of those interim labels. They are being used broadly for peripheral devices, since few companies have reached that stage in the certification process. But deadlines for the interim stage are expected soon.

Garner summed up Tandy's position as he understands it by saying that "All of our product line will eventually meet compliance." The statement sounds positive and reassuring to consumers worried about the obsolescence of their micros. But what has he said? Will the Model I be dropped from the product line? Or will the Model I be remodeled? ■

by Nancy Robertson
80 Staff

Class A Radiation Limits

Frequency (F) (MHz)	Distance (meters)	Field Strength (uV/—)
30-88	30	30
88-216	30	50
216-1000	30	70

Class B Radiation Limits

Frequency (F) (MHz)	Distance (meters)	Field Strengths (uV/—)
30-88	3	100
88-216	3	150
216-1000	3	200

Table 1

Model I—not certified
Model II—*verified* Class A
Model III—not certified, although the application has been filed.
Color Computer—certified as a television interface under separate FCC regulations.

Table 2

Micros in the Campaign

Continued from page 54

contract with a service bureau in Illinois, which did their data processing on an IBM 360. Because of their tight budget, all information traveled to this service bureau in hard copy form by mail, or by telephone.

Boswell agreed that the Anderson campaign was at a disadvantage by lacking the funds to gather demographic characteristics by computer. Any statistics gathering of this sort had to be done manually, through Information Service subscriptions or by contracting with a pollster. The same was true for researching other candidates or local issues in areas in which Anderson was campaigning.

Reagan's Retinue

Several calls to Reagan's headquarters, and conversations with several different people there, produced the information that computers were being used in the campaign, but no one was sure what kind of computers, or what applications, or who was in charge.

So there you have it computer fans. Computers were involved in nearly every aspect of presidential campaign planning this year. Yet, unanswered questions remain: Does the handicap of not being able to afford all the time-saving and speech-directing applications seriously affect chances of winning an election? If you don't know how your computers are used, or where they're kept, can you really use them effectively? How much of what we saw in 1980 was acutally a data bank's vision of how to deal with the opponent's latest political speech? Computers may not have written the speeches in this election, but what about next time? ■

by Debbie Marshall
80 Staff

Compucrime

Continued from page 52

the room where the computer is housed; 2) protecting the medium upon which the program is stored; 3) protecting the medium upon which the data is stored; and 4) controlling the forms on which the output is printed.

A more effective type of security measure involves the software itself—such as the use of passwords which cause the program to abort unless specific, prearranged information is input upon request.

Another common method of protection is encryption. There is now a data encryption standard (DES), a chip for implementing this standard, and lots of proposals for alternative systems, including some very attractive "public key" systems.

Aside from the technological aspects of controlling computer crime, the need to develop more effective psychological/motivational techniques is also being popularized.

Deterrents and controls notwithstanding, the fact remains that the vast new frontiers now opening up through the applied genius of microprocessors continue to attract the outlaw element who, like crooks of every era, thrive in an environment where controls have not yet caught up with the expansion. ■

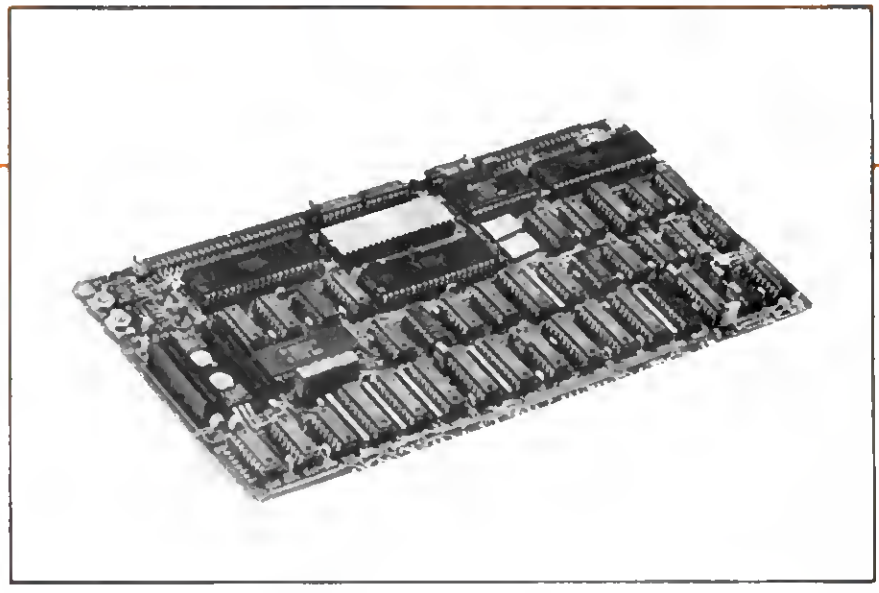
by Paul Quinn
80 Staff

Educational Software Symposium

An Educational Software Symposium will be held Jan. 17-18, 1981 at the Holiday Inn, Bridgeport, CT. Topics will include "Educational Software for Elementary Schools" and software for particular curriculums, as well as how to write educational software. Registration is $85. Contact Queue, 5 Chapel Hill Dr., Fairfield, CT 06432 for reservations or further information. ■

NEW PRODUCTS

Edited by Chris Crocker



MSC Processor Board

S-100 Processor Board Eliminates Polling

The Model CPD-280 is a Z-80A based, second-generation processor board designed for the S-100 computer bus. It operates at four megahertz and is geared toward multi-user systems. Eight vectored priority interrupts maximize the central processor's executable time by eliminating the need for polling. A real-time clock generates the interrupts required by the multi-user operating system.

Two serial and two parallel ports utilize direct memory access for high speed data transfer. All functions are performed by LSI chips.

The second generation processor board costs $750. Volume discounts are available from Measurement Systems and Controls, 867 N. Main St., Orange, CA 92668.

Reader Service ✓ 164

Double Density Software

Disk Zap 2.3, a disk editor from Micro Systems Software will work either single or double density disks. It is track and sector oriented, and offers access to all parts of the disk. It formats and backs up disks, as well as edits them.

DOSPLUS 3.1D, also from Micro Systems, is similar to most single density operating systems, but offers the increased storage of double density.

Disk Zap 2.3 costs $19.95, and DOSPLUS 3.1D is $99.95 from Micro Systems Software, Inc., 5846 Funston St., Hollywood, FL 33023.

Reader Service ✓ 172

Utility Cleans Disks

Nupurge is a utility program that cleans disks of unwanted clutter after a program is killed. It loads the disk directory into memory, and lets the operator choose which programs to keep and which to kill. The unused sectors are zeroed.

In addition, according to Soft Sector, Nupurge will figure out the password of any program. The program costs $24.95 on disk from Soft Sector Marketing Inc., P.O. Box 2471, Livonia, MI 48150.

Reader Service ✓ 173

Education Sampler

Education Sampler is a program for high school math/science courses. It will test, self-drill, or provide answers in three subject areas: algebra, geometry, and chemistry.

The user may select an answer accuracy level for testing purposes between .01 and 5% error. The cassette version costs $15 from Harry H. Briley, P.O. Box 2913, Livermore, CA 94550.

Reader Service ✓ 166

Radio Shack 1981 Computer Catalog

Radio Shack's 1981 TRS-80 Computer Catalog No. RSC-4 lists Model I and II equipment, as well as the new Model III, Color Computer and Pocket Computer.

Also notable are the Daisy Wheel II Printer which produces typewriter quality hard copy for $1,960; a Plotter/Printer that produces hard graphics for $1,460; and Videotex, a two-way information retrieval system terminal for $399. New educational hardware for the TRS-80 includes the Network I Controller, which allows teachers to upload and download programs for up to 16 student stations for $499.

The catalog also lists books and software and is available free from Tandy/Radio Shack, 1300 One Tandy Ctr., Fort Worth, TX 76102.

Reader Service ✓ 185

Business Analysis And Forecasting Package

Oracle-80 is a business analysis and forecasting package from Instant Software. The package can be used in sales analysis and forecasting, product planning and business planning. Investors can analyze stocks, company trends and growth rates. The package can be used in analysis of general economic climates, business cycles and energy consumption trends.

Oracle-80 requires a TRS-80 Level II with 16K and a disk drive, and costs $99.95 for disk or $75 for the cassette version.

Oracle-80 was released in Instant Software's fall-winter catalog. The catalog includes 55 new programs for the TRS-80

Software for any season.

At The Bottom Shelf, we're continuing to produce some of the best TRS-80™ software available anywhere. In the two years since we released the Library 100, we've developed sophisticated data managing, general accounting, and system utility packages. We also developed the first disk drive head cleaners for both Model's I and II. The result has been resounding acclaim from users, dealers, and computer magazines.

But this is just the beginning. In 1981, TBS will introduce for the Model II the most dazzling and intricate applications software it has yet produced. The culmination of ten months of work. In early 1981, you will witness MEGAMAIL, the most thorough and professional mailing system ever written for the Model II.

We've come along way in two years. We are now on the threshold of a new era in computer programming. The Bottom Shelf is leading the way. With software for all seasons.

TRS-80™ is a registered trademark of the Tandy Corporation.

TBS™ 6

THE BOTTOM SHELF, INC.

(404) 296-2003 • P.O. Box 49104 • Atlanta, GA 30359

Models I and II, Apple II and PET.

Catalogs are available free from Instant Software, Peterborough, NH 03458.

Reader Service ✓329

Pharmacists' Aid

Pharmacy Associates' catalog lists programs for medical and pharmaceutical use. Programs included are: Antibiotic Dosing, Aminoglycoside Dosing, and Total Parenteral Nutrition.

The programs require TRS-80 Level II or Disk BASIC with 16K. The catalog also lists a TRS-80 Pocket Computer version of Aminoglycoside Dosing. All programs are available from Pharmacy Associates, 1202 Fox St., Bossier City, LA 71112.

Reader Service ✓ 170

Catalog Lists New Books

A 16-page catalog from Creative Computing Press features three new books. *Computers in Mathematics: A Sourcebook of Ideas* offers 224 pages of classroom activities. *The Impact of Computers on Society and Ethics: a Bibliography*, compiled by Gary M. Abshire, lists over 1900 entries, including books, magazine articles, news items, and scholarly papers. *Katie and the Computer* by Fred D'Iganzio and Stan Gilliam is an illustrated adventure story that explains the workings of computers to children.

The catalog also describes a record album of computer music, a board game, T-shirts, reprints, back issues of *Creative Computing* and *ROM* and ten additional books. The catalog is free on request from Creative Computing Press, P.O. Box 789-M, Morristown, NJ 07960.

Reader Service ✓ 160

Shrink Data Files

Reduce is a program designed to reduce the size of a data file made with Radio Shack's Profile data file system. It allows a number of data files to be used on the same disk with a BASIC program.

The program also will reduce the file size on the Profile disk to use only one file in a BASIC program, and use the BASIC program on the Profile disk. Reduce costs $19.95 and is available from Micro Development Systems, 720 Dartmouth Lane, Schaumburg, IL 60193.

Reader Service ✓ 162

Circuit Design Software

The Circuit Design Software programs are 37 engineering and statistical programs on seven cassettes from Howard W. Sams and Co.

The new series of programs are for use in the design of active filters, matching pads, attenuators, heat sinks, integrated circuit timers, Zener diode regulators and bipolar translator circuits. The programs allow the operator to solve simultaneous equations with real and complex coefficients and polynomial roots. The operator also can determine the effects of design perameters.

The packages require Level II BASIC and at least 16K RAM. Prices range from $16.95 to $21.95 and are available from Howard W. Sams and Co., Inc., 4300 W. 62nd St., Indianapolis, IN 46268.

Reader Service ✓ 163

Program Calculates Intoxication

Intoxitron, a program from The Lawtech Co. estimates a subject's blood alcohol content and degree of intoxication, based on sex, weight, number and strength of drinks, and time since the first drink. A single occasion can be analyzed, or a general chart may be produced.

INC., another program from Lawtech, explains cumulative voting, performs calculations necessary to understand and allocate shareholder voting power, and contains a checklist of pitfalls, as well as a bibliography. Each program requires a 16K TRS-80 with Level II BASIC and costs $16. They are sold by The Lawtech Company, P.O. Box 1523, La Grande, OR 97850.

Reader Service ✓ 174

Hard Disk System Works with TRSDOS

HDOS-2 is a hard disk operating system designed specifically for use with TRSDOS 1.2 on the TRS-80 Model II. The program allows a standard Corvus hard disk drive to be interfaced to existing software with minor changes to the software, according to Computer Program Associates.

The system occupies 1K at the top of memory, and allows multiple drives to be used. It restores PEEK and POKE commands, and adds three new BASIC commands. HDOS-2 supports only random access files; and programs or sequential files may not be stored on disk.

Prices were not released. HDOS-2 is available from Computer Program Associates, 150/6 Beltway Dr., Dallas, TX 75234.

Reader Service ✓ 178

Parallel I/O Board Has 5-V Supply

The Parallel Input/Output Board is a new peripheral board from Persteve Electronics, Ltd. for the TRS-80. It connects directly to the edge connector at the back of the computer. The board contains nine eight-bit I/O ports and is controlled via the Level II BASIC instructions INPUT and OUTPUT. It is powered by a single 5-volt power supply.

The assembled version costs $65; an unassembled bare board is also available from Persteve Electronics, Ltd., P.O. Box 3623, Stn. D, Ottawa, Canada K1P 6H8.

Reader Service ✓ 167

General Accounting Package and CP/M System

A General Accounting Package consisting of a general ledger, accounts receivable, accounts payable and a complete CP/M operating system for the TRS-80 Model II are available from Microed.

The package uses double entry with user-definable accounts. Seven levels of account classification are possible with up to four digit fields at each level.

The CP/M operating system included has all of the standard CP/M programs plus Microed-written utility programs. These utility programs can format disks, copy disks, and operate on a single drive. Microed CP/M for the Model II is capable of single or double density operation and automatically senses the density of the disk. The complete package costs $415 from Microed, 3910 Bandini St., San Diego, CA 92103.

Reader Service ✓ 161

Corrections

Regrettably, two photos were interchanged in the November New Products section. The Micromatic 80 belongs on page 58 and the Mediamix 50/80 interface on page 56. Our apologies for the confusion.

Also, we reported the address incorrectly for Multi Media Systems in our September issue. The correct address is Box 41084, Indianapolis, IN 46241.

Enjoying 80 MICRO ? then read on...

80 MICROCOMPUTING has proven, in its first several issues, that it can give you more information on the TRS-80* than any other single source. The magazine has grown more informative with each month and we still have lots more interesting ideas in the works for you.

With the TRS-80* (or 90...etc.) being the most popular microcomputer in the entire world, you are going to benefit from this in many ways. The more computers there are out there of one kind...the more good programs you are going to have for this system. I hope that is obvious. You may be sure that *80 MICROCOMPUTING* will be packed with the shorter programs and reviews of the larger ones. You can waste an awful lot of money on stuff that looks great in the ads, but fizzles out when you try to use it. You need our reviews.

The wealth of programs will also mean that there will be much better programs for the TRS-80* than any other system. Put yourself in the seat of a computer programmer and you'll understand this. If you are going to spend several months developing a comprehensive program, and it takes all of that to write and debug a big program, would you write it for a system which has sold one hundred units or one which has sold over 300,000 systems? The answer is obvious...and this is why we are already seeing programs coming out for the TRS-80* which are far better than anything for any other system on the market. This is tough for other systems ...the law of the computer jungle.

Between our connections with Instant Software, the largest publisher of microcomputer programs in the world, and Kilobaud MICROCOMPUTING, you know that *80 MICROCOMPUTING* is going to be your most important link with software for the TRS-80*.

With Instant Software being sold and promoted in every country in the world where the TRS-80* is being sold, our input of programs is also the best in the world. We get programs submitted from everywhere...often from 50 to 100 a week! You'll get the cream of the crop either published or reviewed in *80*.

HARDWARE TOO

The same law of the computer jungle holds for hardware. Would you, as a manufacturer, market an accessory for a system which has sold 100 units or would you go first for the one which has sold hundreds of thousands. It is, as with software, self-evident why the great bulk of the hardware accessories for computers are for the TRS-80* these days.

80 MICROCOMPUTING has the advantage of the use of the largest and most complete microcomputer lab in the world...the one developed for Instant Software and Kilobaud MICROCOMPUTING. This means that most new pieces of equipment are tested and in use by our staff...and this means that we can tell you what we think is outstanding...and where we find ripoffs. This lab is important to you.

SUBSCRIBE

If you are not already a subscriber to *80 MICROCOMPUTING*, please get signed up right now. The yearly rates are $18, and that is a bargain. Just one single program of use to you can be worth much more than that. One review of an accessory could save you many times that much investment. I would appreciate it if you would appoint yourself a committee of one to get more subscribers for the magazine. You will benefit even more than we do here at the magazine...because the more readers we have, the more ads we will be able to attract...and the more ads, the more pages of articles you will get every month.

The *80* market can, I think, support a couple of hundred pages of ads...and that would mean a magazine of nearly 500 pages a month. That should hold you. You may not have time left to use your computer.

ENCYCLOPEDIA

If you've read Kilobaud MICROCOMPUTING, you know that I try hard not to duplicate published material. My concept is that every reader should save every issue (we sell inexpensive boxes for this so they can sit on your library shelf) and treat the magazine as a continuing encyclopedia of computing. I make sure that much of the material in each issue is written in simple language so it will be understandable by even the rawest newcomer to computers. Oh, I have articles for the more advanced users too, so you'll have something to look back over later and use as your understanding of your system grows.

Try to think of *80 MICROCOMPUTING* as more of a large club newsletter than an ivory tower high-level publication. I'll leave the pomp to other publishers...the ones with the well-deserved inferiority complexes who cater to their inadequacies by publishing esoteric baloney. This magazine is written by the readers and edited by people whose aim is to help you enjoy your TRS-80*.

SAVE

With each issue costing $2.50 at your computer store, that's $30 a year. For $18 a year you can subscribe...at least for now. As the magazine expands, please do not be surprised if the cover price increases, along with the subscription price. I started *73 Magazine* for radio amateurs twenty years ago with a cover price of 37¢ (two for 73¢) and it is up to $2.95 a copy now (and it is the largest of the ham magazines).

For you bargain hunters...and those who find that one year goes by all too rapidly, the three year rate for *80* is $45. This, too, will be going up...reflecting the inflation, paper increases, postage increases, and a short vacation for me in Hong Kong next year. Someone has to pay for that.

*TRS-80 is a trademark of Tandy Corp

If the coupon below has been used, please fill out subscription form on the Reader Service card in the back of the magazine

YES! Sign me on as a subscriber to 80 Microcomputing for only $18 a year!

Card # _____ Exp. _____

Signature _____

Name _____

Address _____

City _____ State _____ Zip _____

- ☐ 12 issues – $18
- ☐ 36 issues – $45
- ☐ Please bill me
- ☐ Payment Enclosed
- ☐ Master Charge
- ☐ VISA
- ☐ American Express

80 microcomputing
Peterborough, N.H. 03458

Subscription begins with next published issue
Back issues, while available are $3 each
Canada: $20 per year US funds
All other foreign subscriptions, $28 one year only.

39DB8

PMC-80

Level II 16K at $645

SOFTWARE COMPATIBLE

- Reads all Level II BASIC tapes
- Reads all SYSTEM tapes
- Full range of peripherals
- Video output for monitor and TV
- Optional FASTLOAD at 8000 baud
- Optional Upper/Lower case

The PMC-80 is a "work-alike" computer to the popular TRS-80* Model 1, Level II by Tandy, Radio Shack. The PMC-80 has 16K bytes of RAM and the complete Level II 12K BASIC ROM by Microsoft that makes it 100% software compatible with programs from Radio Shack and from the hundreds of other independent suppliers. The built-in cassette player reads standard Radio Shack programs for the TRS-80*.

The PMC-80 will operate with any of the many peripherals Radio Shack and other independent vendors have invented to plug into the TRS-80*. Most importantly, the interface Adapter permits Expansion interfaces with memory expansion to 48K to be added. An Expansion Interface will also permit the addition of Radio Shack compatible 5¼" disks and disk operating systems, RS 232, printers, etc.

Sold through computer stores.

*TRS-80 is a registered trademark of Tandy, Radio Shack.

Personal Micro Computers, Inc.

↙422

475 Ellis Street, Mountain View, CA 94043 (415) 962-0220

Payroll System Maintains Tax Files

PR is a payroll system for the TRS-80 Model II that calculates payroll for employees while maintaining monthly, quarterly and yearly totals for reporting purposes to multiple states. Tax tables are maintained via on-line commands with no programming required, according to Micro Architect, Inc.

PR requires TRSDOS 1.2, a 132-column printer, a dual disk system and 64K memory. The program costs $129 from Micro Architect, Inc., 96 Dothan St., Arlington, MA 02174.

Reader Service ↙ 181

Index on One Disk or Two

Two new versions of the Keyword Indexing System are available from Northeast Microware. The Keyword Indexing package is a series of programs enabling the user to create a disk file, build an index of all key words, and search for them using combinations of key words.

The new systems include an enhanced version for two disk systems and a compressed version for one disk systems. Both require 32K of memory and run under TRSDOS. They are available from Northeast Microware, P.O. Box 2133, Boston, MA 02106.

Reader Service ↙ 183

Lighting and Fault Current Programs

Two electrical engineering programs from MC.2 have on-board files of equipment and fixture characteristics.

The E3M Fault Current Program uses a per-unit calculation procedure and permits an unlimited number of bus voltage levels, panels and branches. Three-phase symmetrical voltage and fault currents are calculated at any point in the system, with or without line voltage drop.

The E5M Lighting Program automatically calculates the number, spacing and location of luminaires required to give a desired level of illumination in a project of up to 100 rooms. The program also will determine the lighting level supplied by a given number and type of fixture.

Prices were not released. The programs are written for the TRS-80 Models I and II from McClintock Corp., P.O. Box 430980, Miami, FL 33143.

Reader Service ↙ 185

Belden Bit Driver

Short-haul Modem

The Belden Model 9338 metallic conductor Bit Driver short-haul modem is part of an RS-232C compatible data transmission system.

The Model 9338 metallic Bit-Driver modem provides asynchronous simplex and duplex data transmission. The metallic conductor unit is recommended for use in clean electrical environments. Depending on the type of cable selected, operation range extends from 1500 to 4500 meters. The price of Model 9338 is $195 from Belden Corp., 200 S. Batavia Ave., Geneva, IL 60134.

Reader Service ↙ 184

Program Tests, Drills

T.E.S.T. is a classroom aid from T.Y.C. Software. The package contains two programs: a Maintenance Program and a Test and Drill program. The Maintenance Program creates a test of up to 35 questions and saves it on cassette. In order to produce a test, a question is typed on any topic (up to 240 characters), the type of question—true or false, multiple choice, or completion—and the correct answer entered. When finished, the test is saved on cassette.

Test and Drill is a utility program designed to accept the test prepared by the maintenance program. With the Test and Drill program, students can either use the questions as a review, take a scored test, or the teacher can have the computer prepare a printed test or worksheet with answer key.

The package contains two programs and a manual for TRS-80 Level II, 16K for $11.95. For more information, contact T.Y.C. Software, 40 Stuyvesant Manor, Geneseo, NY 14454.

Reader Service ↙ 176

Terminal Programs Transfer Files

SMART80E and SMART80C are terminal programs for use with the Exatron Stringy Floppy and cassette-based systems, respectively. The programs are used in conjunction with a direct-connect telephone interface called The Microconnection.

The terminal programs allow the transfer of BASIC programs and source code files. The programs also feature software selection of half and full duplex plus the ability to transfer text created by either Electric Pencil or Scripsit in upper/lowercase. For additional information on SMART80E, SMART80C or The Microconnection, contact The Microperipheral Corp., Box 529, Mercer, Is., WA 98040.

Reader Service ↙ 177

COBOL Compiler On Release 2 CP/M

RM/COBOL, a high-intermediate level ANSI-74 COBOL compiler, is available on Release 2 CP/M systems for $495.

This compiler, compatible with several minicomputer COBOL compilers, has alternate keys (multi-key ISAM), CRT screen handling, program segmentation, interactive debug, and other Level II features. Implemented under the Cybernetics, Inc. version of Release 2 CP/M on the TRS-80 Model II, RM/COBOL is source-program compatible with Tandy's COBOL.

The RM/COBOL User's Guide and the RM/COBOL Language Manual may both be obtained for $40 (refundable upon purchase of RM/COBOL), from Cybernetics, Inc., 8041 Newman Ave., Suite 208, Huntington Beach, CA 92647.

Reader Service ↙ 169

Five DOS Utilities And Teachers' Package

The Alternate Source has a five-program utility package for the TRS-80. Three of the programs are written in Z-80 machine language and can be used with either Level II cassette systems or with DOS systems. Two are written in BASIC, for use with DOS systems only.

The three Z-80 utilities ere distributed with a relocatable module which allows them to be dumped at the user's specified starting address. They are BTrace, Compress Program Utility and Search.

When TRON is activated, BTrace leaves the screen display intact and places any lines being executed in the upper right-hand corner of the screen. Compress Program Utility allows BASIC programs to be compressed in a variety of ways. Search will locate eny BASIC line containing whatever argument the user wishes to find.

The two BASIC DOS programs are Changes and Replace. Changes provides a screen or printed listing of the differences between two programs. Replece will locate ell occurrences of en argument and replece it with a string.

The package is available on a single disk for $29.95.

Schoolmaster, a separate package, is e record keeping system for teachers. It generates cumulative reports for each student, and flags students whose assignments are missing. Teachers can examine a variety of grading methods before recording grades, according to the Alternate Source. Schoolmaster will present both individual and class statistical data.

Schoolmaster requires a 32K TRS-80 with one drive. The program comes on diskette for $24.95. Programs are available from The Alternate Source, 1806 Ada Street, Lansing, MI 48910.

Reader Service ↙326

Disk File Directory

Master Diskette Directory version 1.1 reads, stores and categorizes the directories of up to 320 disks. The program will list all files on disk, by file extension, disk number or program category.

Master Directory will also search for a file name and list every number of that file, its size, and the number of the disk containing the file.

The program is available for $29.95 from Micro Systems Software, Inc., 5846 Funston St., Hollywood, FL 33023.

Reader Service ↙341

Information Retrieval for TRS-80

SE (Search Entry) is a general purpose information retrieval program. It is a machine language program for the TRS-80 Model I, Level II.

SE's command structure facilitates data entry, data searches, and quick data storage and retrieval on tape or disk, according to the manufacturer, Information Technology Systems. Some commands are available from the ENTER Option response with a single keystroke.

Targets can be any combination of 64 characters, employing unlimited ANDs and ORs, according to ITS. The program includes error messages and error checking procedures.

Data entries are identified by a three-character code assigned by the user. All of memory, less 4K for the program, is available for storage.

SE is sold in two versions: SE2.0 for 16K Level II (cassette) costs $24.95, and SE3.0 for DOS up to 48K costs $49.95. SE is available from Information Technology Systems, Post Office Box 2667, Saresota, FL 33578.

Reader Service ↙334

Series of Educational Instruction and Utilities

Rite 80 Software is selling several series of field tested programs for use in schools. Written for Level II machines, the series are Math, Spelling, Topics, Earth and Rollbook.

The Math Series consists of three programs for individual or group work, designed to help students increase their speed and accuracy in basic arithmetic. The three programs in the Spelling Series drill students on rote memorization of spelling words.

The Topics Series, four programs, allows teachers to test students on any subject, using short phreses or single words as answers. The program will accept different words with the same meaning for correct answers.

Earth is a video enimetion of the earth rotating on its axis. Rollbook is a disk utility for teachers. It will record up to 100 grades for 40 students.

Rollbook is priced at $49.95 from Rite 80 Software, 4660 Willens Ave., Woodland Hills, CA 91264. The other series are priced by program. Programs cost $19.95 each, with discounts given for the purchase of en entire series.

Reader Service ↙335

Elcompco Disk Drive System

The Elcompco disk drive is a case and power supply with either MPI-B51 or Shugart SA-400 drives. A large 18,000 uF capacitor and fixed voltage regulators are included to reduce ripple and noise from the power supply. The heat sink is mounted externally, and allows the drive system to run cool while powering drives, according to Elcompco.

The system is available with one drive or two. Kits are available for the case and power supply only, or including drives. The drive will power mini-floppy drives compatible with Shugart or MPI power requirements.

Dual drives with case and power supply cost $800. Single drive in double case is $475, and a single drive in single case costs $400. The kit without drives costs $135.

The drive systems were released in Elcompco's winter catalog of hardware end software.

Catalog and disk drives are available from Elcompco Microcomputer Peripherals, P.O. Box 6133, Albany, CA 94706.

Reader Service ↙339

TRS-80 Data Management System

Data Access Corp. has DataBank software for TRS-80 Model II microcomputers. Databank is a system of pre-programmed, data independent modules that are adaptable to each user's requirements.

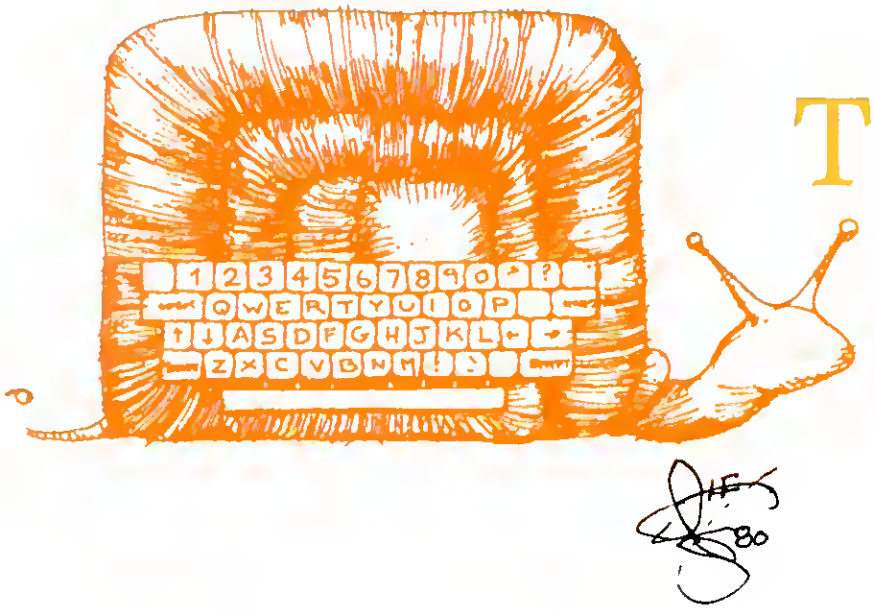
File maintenance, data management and report generation functions are operable as soon as the user indicates file specifications. Typical applications include mailing list maintenance, inventory and accounting records, student or personnel files, and patient/client data systems.

DataBank uses hashing, assembler subroutines and other techniques. Files can span up to four disk drives with as many as 32,767 records. Key access time to a given record is a second or less.

Modules are divided into four main groups: configuration utilities, file maintenance, report generator, end e subroutine library. A multi-purpose editor program is also included.

DataBank runs under TRSDOS and BASIC. It is priced at $249 per installation from Data Access Corp., 4221 Ponce De Leon Blvd., Coral Cables, FL 33146.

Reader Service ↙175



The Dvorak Keyboard

Waldo T. Boyd
Jon Etherton
PO Box 86
Geyserville, CA 95441

You have just taken possession of a dream—a new TRS-80. Set it up, switch it on, and reach for the home row on the keyboard. Your fingers flick from key to key, and, slowly but surely, into the memory goes your first program. *Wunderbar!*

The chips work perfectly; the terminal screen glows brightly with your first effort. Everything seems to be in order. Could anything be wrong?

Nothing, you say. It's perfect!

Let's look more closely. What about the arrangement of the alphabet on the keypads: Q W E R T Y U I O P. Just like the millions of typewriters in homes and offices everywhere in the English-speaking world, right?

Yes, it is. And that is precisely what is wrong with hundreds of thousands of computers. The computer itself is lightning-fast and potentially, error-free, except for the most vital consideration of all —the man to machine interface.

The letters on the keyboard are placed in positions that will cause you to make errors in your input, that will tire you, that will keep your entry speed far below your true operating potential.

A Little History

Your computer inherited a keyboard that was originally designed for the typewriter by an inventor named Christopher L. Sholes in 1873. His first production machine looked very much like a sewing machine. Its typebars were hidden from the operator's view, striking upward from beneath the platen (paper roller), so the operator couldn't see what had been typed until three or four lines later.

Worse than that, typebars, which lay next to one another, had so much mutual friction that if the operator struck an adjacent key too soon, the first key struck would fail to fall back into its rest position in time to miss the upcoming key. The result was a key jam, not easily remedied in those early hand-built machines.

Sholes was no less than ingenious in his approach to fixing these jams. By studying the frequency of occurrence of the letters in the majority of common words in the English language, Sholes reduced the number of jams per sitting.

He found that by placing the operating keypads in a certain sequence, he could slow down the faster operators, and the typist could hunt-and-peck through dozens of letters with no more than one or two serious jams. This sequence ensured that the operator stayed below ten words per minute, the critical speed of his machine. This, we can say with 20-20 hindsight, was human engineering—in reverse.

Look once more at the keyboard on your TRS-80: You are viewing the keyboard Mr. Sholes produced for the specific purpose of slowing down the 1870's operator, so his machine would operate without jamming!

It wasn't long before far better machines than Sholes' cumbersome "sewing machine" were developed and marketed, but for some inexplicable reason every manufacturer who jumped into the burgeoning typewriter market copied the keyboard laid out by Sholes. Everyone took for granted that QWERTY was as good as any other arrangement.

Bruce Bliven, Jr., author of *The Wonderful Writing Machine*, has this to say about QWERTY: "Judged scientifically . . . from the standpoint of the touch typist, this arrangement of the alphabet is madly inconvenient. According to one of the many persons, including psychologists, engineers and student Ph.Ds who have studied it, the standard keyboard is considerably less efficient than if the arrangement had been left to simple chance."

A Breakthrough

But for the perseverance and insight of Dr. August Dvorak, late professor of English at Washington State University, we would be stuck with QWERTY for all time. Dvorak heard the anguished cries of a few far-sighted touch-typists and arranged a

"When he had completed his work and tested it in the early 1940s, his keyboard was found to be twenty times easier to use..."

U.S. Navy contract to humanly-engineer the typewriter keyboard.

When he had completed his work and tested it in the early 1940s, his keyboard was found to be twenty times easier to use than QWERTY.

His brain-child, the Dvorak Simplified Keyboard (DSK), can be learned in one-quarter to one-half the time required to learn Sholes' old system, and DSK touch typists become so proficient that they leave the QWERTY typist far behind in speed and accuracy.

DSK users commonly type 100 wpm, while the average QWERTY typist is hard-put to better half that speed. The world typewriter speed record holder is a DSK typist.

Why, if the DSK is so efficient, doesn't industry adopt it and build typewriters and computers with the new keyboard? The answer lies in two realms that affect mankind universally: tradition and economics. Typists train on QWERTY in high school. They enter the job market, able to type about 45 wpm on the average. Their prospective employers have QWERTY keyboard typewriters waiting for them. The schools provide typists to fit the business office; typewriter manufacturers provide machines to fit the operators who are trained by the schools. Catch 22!

Dr. Dvorak's keyboard was ready for market in 1944, more than a decade ahead of ENIAC, the first major computer. Yet, the first computer operators were typists who had mastered the QWERTY keyboard. Not surprisingly, these highly accurate, lightning-fast machines are today provided with—nay, *saddled* with—a QWERTY keyboard!

The DSK however, is finally catching on, in spite of tradition in the marketplace. A few pioneering court reporters and free-lance writers have used the DSK over the years. In the past twenty years the number of users has increased to the extent that two typewriter manufacturers now offer the DSK on new machines at no extra cost: IBM and Smith-Corona (SCM).

So far, no manufacturer has offered the DSK as an option on new computers, but the computer itself provides a simple way to use either keyboard at will. Software for this purpose has been developed and will soon be available on tape.

The Software Route

The software program (Program Listing 1) presents a reasonably simple way to convert the TRS-80 keyboard from QWERTY to DSK. This program permits instant changeover from QWERTY to DSK and back to QWERTY by pressing two keys simultaneously, the shift key and the zero key. Thus, instant comparison of one keyboard with another is possible.

"Why, if the DSK is so efficient, doesn't the industry adopt it and build typewriters and computers with the new keyboard?"

The 183-byte program is loaded into the desired memory location by means of a BASIC relocating program, which also gives some operating instructions and does decimal to hexadecimal conversions of the starting and ending addresses. The machine language code is contained almost entirely in the data statements on lines 170-184. (If you want to save some effort you can leave out all but Lines 8, 100-2330 and 10000-10040.)

The program is compatible with other machine language utilities, such as Radio Shack's KBFIX, but KBFIX must be loaded first.

To load the program below another machine language program (such as a printer driver) subtract 183 from the current memory size. This gives you the new memory size to be used when powering up the computer.

Be sure to tape at least one copy of the BASIC program before running it. If any of the data statements contain an error the Z-80 may jump back to memory location zero, thus wiping out your program.

After loading the BASIC program, type RUN. The program asks you if you remembered to set memory size and then prints some explanatory information before asking for the starting address. This is usually, but not necessarily, the same as the memory size.

After you enter the starting address, the program proceeds to POKE in the memory locations selected. These locations and data are displayed on the screen.

The program then prints more information, including the starting and ending addresses in hexadecimal. You may want to make note of the address for the start of the lookup table, for future reference.

To execute the program, type SYSTEM (enter). When you get the *? prompt, type / and the starting address (in decimal). The keyboard should now be in DSK mode. If an unexpected response appears, such as the memory size question after executing the program, reload your tape (or the program), check the BASIC program (especially the data statements), and try again.

Check to see that the keyboard is indeed in DSK mode by typing asdfg. The letters aoeiu should appear on the screen. If they don't, press the shift and zero keys (QWERTY position) down simultaneously.

The BASIC program, now having done its job, can be cleared from memory by typing NEW. The machine language ob-

Apparat introduces

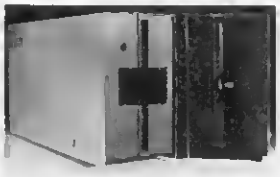
More bytes per buck

405 K/bytes of storage. Apparat has combined its Newdos/80 operating system and a dual-sided 80 track mini-floppy drive to give you 405,000 bytes of storage in a single volume. Modification patches to Newdos/80 expands the capability of single density drives, so you'll have greater applications for your TRS-80 model 1.*

Drives plug directly into an expansion interface with no modification required so you can now have over 1 megabyte of storage on-line with standard mini-floppy diskettes. Each drive has 316 free grans, for a total of 948, on a maximum of three 80 track drives, which can be added to a TRS-80.

Upgrading to double density is possible by running under most double density controllers.

And, you can choose either an MPI or Tandon Drive Mechanism. Drives come complete with case, power supply, interface cable and documentation including patches to Newdos/80. Either drive mechanism is priced at only $839 with additional drives available at $789. At 482 bytes per buck, it just might be the answer to your storage problems.



MICRO CLUB CHARTER MEMBERSHIPS

How would you like to be the first to obtain new applications for your TRS-80® and ***save money?***

Then the Micro Club is for you!

A vast array of software, especially games, exists for your microcomputer. But software alone can only partially fulfill the power and usefulness of your computer.

You probably own a disk and a printer—hardware which also enhances the usefulness of your computer. Other than these common peripherals, relatively little hardware exists in comparison to software, and most of the hardware available is prohibitively expensive.

The MICRO CLUB has been established to meet this need for a variety of inexpensive hardware devices—tools which will help you get more satisfaction from your computer.

Each month the MICRO CLUB will announce one or more new hardware products complete with complementary software and comprehensive documentation. Some products to be offered in the coming months include a date/clock event timer, a PROM Burner and an energy efficiency measuring instrument. Best of all, MICRO CLUB members will usually pay less than $100 for products which eventually will retail for $150 or more.

Membership in the MICRO CLUB costs only $10 for 12 issues of our new products newsletter describing our product of the month as well as other hardware and software specials available exclusively to MICRO CLUB members. Even the membership fee is fully refundable, i.e., you will receive a $10 discount on your first selection. Although there's no obligation to purchase any product, you need make only two selections during the 12 month period to have your membership automatically extended for another year.

The MICRO CLUB will make an excellent present. Simply fill in the coupon and mail to MICRO CLUB.

Please register me as a Charter Member of the Micro Club and send me newsletters describing your next 12 monthly selections. I understand that my membership fee of $10 will be refunded against my first selection and that I need only make two selections a year to remain a member.

NAME _____

STREET _____

CITY _____ STATE _____ ZIP _____

PROCESSOR: TRS-80® _____ APPLE® _____

_____ Check Enclosed ($10.00)

_____ VISA _____ Mastercharge

Card # _____ Exp. Date _____

Mail to: ✓264

MICRO CLUB
4401 S. Tamarac Parkway
Denver, CO 80237

Micro Club is an affiliate of Apparat, Inc., specializing in hardware and software

For additional memberships, send along the above information on a separate sheet. For further details call 303-741-1778.

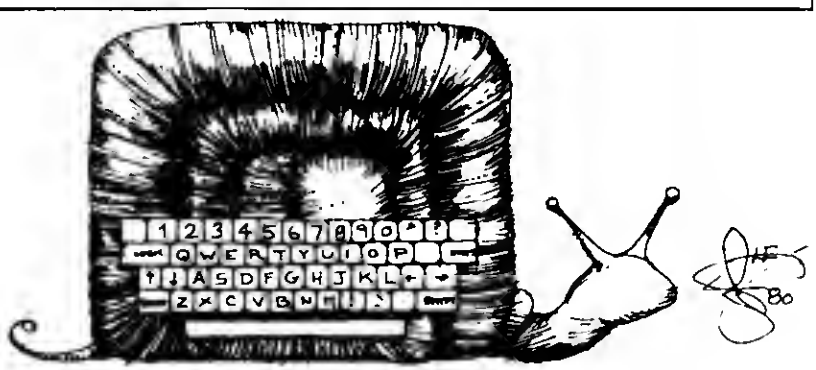
✓Reader Service—see page 274

```
5 CLS
6 CLEAR 30
10 INPUT "DID YOU SET MEMORY SIZE";A$:IF LEFT$(A$,1)="Y
    " GOTO 20
15 PRINT "MEMORY SIZE SHOULD BE AT LEAST 103 BYTES BELO
    W TOP OF MEMORY."
18 PRINT "SET MEM SIZE AND RELOAD PROGRAM.":END
20 CLS
22 PRINT"D V O R A K   S I M P L I F I E D   K E Y B O
    A R D"
23 PRINT"JON ETHERTON  S. 222 ELM #2  SPOKANE, WA  9920
    4"
25 PRINT:PRINT"THIS BASIC PROGRAM LOADS A MACHINE LANGU
    AGE
30 PRINT"KEYBOARD CONVERSION PROGRAM INTO MEMORY LOCATI
    ONS SPECIFIED
40 PRINT"BY THE USER.
50 PRINT"ONCE THE PROGRAM IS ACTIVATED, THE KEYBOARD OF
    YOUR TRS-80
60 PRINT"CAN BE CHANGED FROM 'QUERTY' TO DSK WITH A SIN
    GLE KEYSTROKE"
70 PRINT "PROGRAM SIZE IS 103 BYTES"
98 PRINT"STARTING ADDRESS IS USUALLY THE SAME AS MEMORY
    SIZE."
95 PRINT"IF OVER 32767, THE ADDRESS WILL BE CONVERTED T
    O A NEGATIVE NO.
100 INPUT"WHAT IS STARTING ADDRESS (IN DECIMAL)";ST
105 'IF ST<19096 THEN PRINT"MUST BE OVER 19096":GOTO 10
    0
106 IF ST>65312 PRINT"MUST BE LESS THAN 65312":GOTO 100
109 IF ST>32767 THEN S= -1*(65536-ST):ELSE S=ST
120 FOR A=S TO S+103
130 READ D
140 PRINT A,D
145 POKE A,D
160 NEXT A
170 DATA 42,22,64,34,0,0,33,0,0,34,22,64,195,25,26,205,
    0,0,245,58,16
172 DATA 56,254,1,32,15,50,120,56,254,1,32,0,58,0,0,198
    ,120,50,0,0
174 DATA 50,0,0,254,0,40,2,241,201,241,79,6,0,33,0,0,9,
    126,201,0
175 DATA 0,1,2,3,4,5,6,7,0,9,10,11,12,13,14,15,16,17,10
    ,19,20
176 DATA 21,22,23,24,25,26,27,20,29,30,31
177 DATA 32,42,34,35,36,37,30,39,40,41,33,115,07,45,06,
    90
178 DATA 54,50,55,53,51,49,57,48,50,52,56
179 DATA 03,119,61,110,122,64,65,88,74,69,46,85,73
100 DATA 68,67,72,84,70,77,66,02,76,63,80,79,09,71,75,4
    4,01,70,59,91
102 DATA 92,93,94,95,96,97,120,106,101,62,117,105,100,9
    9,104,116,110,109
184 DATA 98,114,108,47,112,111,121,103,107,60,113,102,4
    3
2000 REM ADDRESSES AND DATA TO BE CHANGED ON RELOCATION
2020 N=ST+16
2040 GOSUB 10000
2060 POKE S+4,LSB:POKE S+5,MSB
2000 N=ST+15
2100 GOSUB 10000
2120 POKE S+7,LSB:POKE S+0,MSB
```

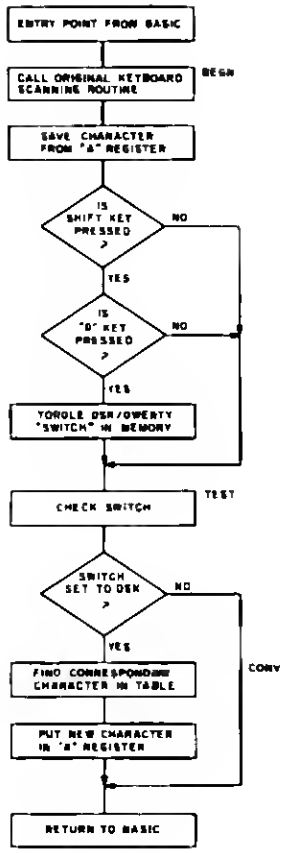
Program continues

```
2140 N=ST+60
2160 GOSUB 10000
2180 POKE S+34,LSB:POKE S+35,MSB
2200 POKE S+39,LSB:POKE S+40,MSB
2220 POKE S+42,LSB:POKE S+43,MSB
2240 N=ST+61
2260 GOSUB 10000
2280 POKE S+55,LSB:POKE S+56,MSB
2320 PRINT "PROGRAM RELOCATED"
2330 E=A-1
2350 PRINT"ENDING ADDRESS IS ";E
2360 PRINT"TO EXECUTE, TYPE 'SYSTEM', PRESS ENTER, AND
     TYPE "
2370 PRINT"'/";ST;"'.  PRESSING THE SHIFT KEY AND '0' W
     ILL CHANGE
2380 PRINT"THE KEYBOARD FUNCTION."
2390 PRINT"THE LOOKUP TABLE BEGINS AT ";S+61;".  ANY KE
     Y FUNCTION MAY
2400 PRINT"BE CHANGED BY POKING IN A DIFFERENT ASCII VA
     LUE."
2420 PRINT"TO MAKE A SYSTEM TAPE WITH T-BUG, TYPE:
2430 N=ST:GOSUB 20000
2440 S$=H$
2450 N=ST+153:GOSUB 20000
2460 E$=H$
2470 PRINT"P ";S$;" ";E$;" ";S$;"  FILE NAME"
9999 END
10000 REM SPLIT POSITIVE-SIGNED ADDRESS INTO MOST AND L
      EAST SIGNIFICANT BYTES
10010 MSB=INT(N/256)
10020 LSB=((N/256)-MSB)*256
10040 RETURN
20000 'DECIMAL TO HEX CONVERSION
20010 A=N/4096
20020 D(1)=INT(A)
20030 B=(A-D(1))*16
20040 D(2)=INT(B)
20050 C=(B-D(2))*16
20060 D(3)=INT(C)
20070 D=(C-D(3))*16
20080 D(4)=INT(D)
20095 H$=""
20100 FOR I=1 TO 4
20115 IF D(I)<10 THEN H$=H$+(CHR$(D(I)+40)) ELSE H$=H$+
      (CHR$(D(I)+55))
20140 NEXT I
20150 RETURN
```

Program Listing 1.



✓ Reader Service—see page 274



Flow Chart

ject code remains in protected memory.

To save the program for future usa, make a aystem tape to load tha machina code program directly. To do this, first load tha program as described. It is not necessary to executa it. Write down the haxadecimal starting, ending and execution addrasses which era displayed on your screen. (Starting and execution addrasses ara idantical.)

Then load a monitor program such as T-Bug. When you get T-Bug's * prompt, load a blank tape in tha recordar and type P and the starting, ending and executing addresses and DSK. DSK is tha fila nama —or use anything you prafer up to six characters. Then press antar.

The tape should taka only a few seconds to record. You might be wisa to maka several dumps of tha program on the sama tape: Simply ratype tha above lina to start aach dump.

The assembly languaga listing (Pro-

"The TRS-80 gets information from the keyboard by scanning eight memory locations and decoding the data into ASCII codes representing each character."

```
4DBC            00010           ORG     4DBCH
4DBC 2A1640     00020 STRT      LD      HL,(4016H)  ;DCB KEYBOARD ADDRESS
4DBF 22CC4D     00030           LD      (BEGN+1),HL ;INTO CALL STATEMENT
4DC2 21CB4D     00040           LD      HL,BEGN     ;NEW DRIVER ADDRESS
4DC5 221640     00050           LD      (4016H),HL  ;INTO DCB
4DC8 C3CC06     00060           JP      06CCH       ;BACK TO BASIC
                00070 ;
4DCB CD0000     00080 BEGN      CALL    $-$         ;ORIGINAL KEYBOARD ROUTINE
4DCE F5         00090           PUSH    AF          ;SAVE KEYBOARD CHARACTER
4DCF 3A1038     00100           LD      A,(3810H)   ;STROBE SHIFT KEY
4DD2 FE01       00110           CP      1           ;SHIFT PRESSED?
4DD4 200F       00120           JR      NZ,TEST     ;SKIP IF NOT PRESSED
4DD6 3A0038     00130           LD      A,(3800H)   ;STROBE ZERO KEY
4DD9 FE01       00140           CP      1           ;ZERO PRESSED?
4DDB 2008       00150           JR      NZ,TEST     ;SKIP IF NOT PRESSED
4DDD 3AF84D     00160           LD      A,(STAT)    ;DSK OR QWERTY STATUS
4DE0 C600       00170           ADD     A,00H       ;TOGGLE STATUS
4DE2 32F84D     00180           LD      (STAT),A    ;STORE STATUS
4DE5 3AF84D     00190 TEST      LD      A,(STAT)    ;GET STATUS
4DE8 FE00       00200           CP      0           ;KEYBOARD IN DSK MODE?
4DEA 2002       00210           JR      Z,CONV      ;IF YES, CONVERT TO DSK
4DEC F1         00220           POP     AF          ;RESTORE NORMAL CHARACTER
4DED C9         00230           RET                 ;RETURN TO BASIC PROGRAM
4DEE F1         00240 CONV      POP     AF          ;RESTORE NORMAL CHARACTER
4DEF 4F         00250           LD      C,A         ;SET UP INDEX
4DF0 0600       00260           LD      B,0
4DF2 21F94D     00270           LD      HL,TBLE     ;TABLE START ADDRESS
4DF5 09         00280           ADD     HL,BC       ;FIND CHAR IN LOOKUP TABLE
4DF6 7E         00290           LD      A,(HL)      ;GET DSK CHARACTER
4DF7 C9         00300           RET                 ;RETURN TO BASIC PROGRAM
4DF8 00         00310 STAT      DEFB    0           ;KEYBOARD MODE (0 OR 00H)
4DF9 00         00320 TBLE      DEFB    0           ;BEGIN LOOKUP TABLE
4DFA 01         00330           DEFB    1           ;(TABLE IS 122 BYTES LONG)
4DFB 02         00340           DEFB    2
4DBC            00350           END     STRT
00000 TOTAL ERRORS
```

Program Listing 2.

grem Listing 2) is intended mainly to explain the program. It is generally best to load the program with the BASIC routine.

How It Works

The TRS-80 gets information from the keyboard by scanning eight memory locations and decoding the data into ASCII codes representing each character.

The keyboard scanning program in ROM is set up as a subroutine which is called by BASIC continuously. The starting address of the subroutine is loaded into a reserved area of RAM called the device control block (DCB) each time the computer is turned on.

There are three DCBs—one each for the printer, keyboard and screen. By loading

Continued to page 78

DEC CODE	KEY	NEW CHAR	ASCII CODES DEC	HEX
0	none		0	00
1	break	same	1	01
2-7	none		2-7	02-07
8	l/c back arrow	same	8	08
9	l/c rt. arrow	same	9	09
10	l/c down arrow	same	10	0a
11	none		11	0B
12	none		12	0C
13	enter	same	13	0D
14-23	none		14-23	0E-17
24	u/c back arrow	same	24	18

Table continues

25	u/c rt arrow	same	25	19
26	u/c down arrow	same	26	1A
27	u/c up arrow	same	27	1B
28-30	none		28-30	1C-1E
31	clear	same	31	1F
32	space	same	32	20
33	!	*	42	2A
34	"	same	34	22
35	#	same	35	23
36	$	same	36	24
37	%	same	37	25
38	&	same	38	26
39	'	same	39	27
40	(	same	40	28
41	)	same	41	29
42	*	!	33	21
43	+	S	115	73
44	,	W	87	57
45	-	same	45	2D
46	.	V	86	56
47	/	Z	90	5A
48	0	6	54	36
49	1	:	58	3A
50	2	7	55	37
51	3	5	53	35
52	4	3	51	33
53	5	1	49	31
54	6	9	57	39
55	7	0	48	30
56	8	2	50	32
57	9	4	52	34
58	:	8	56	38
59	;	S	83	3H
60	<	W	119	77
61	=	same	61	3D
62	>	V	118	76
63	?	Z	122	7A
64	@ (unshifted)	@	64	40
65	A	A	65	41
66	B	X	88	58
67	C	J	74	4A
68	D	E	69	45
69	E	.	46	2E
70	F	U	85	55
71	G	I	73	49
72	H	D	68	44
73	I	C	67	43
74	J	H	72	48
75	K	T	84	54

Table continues

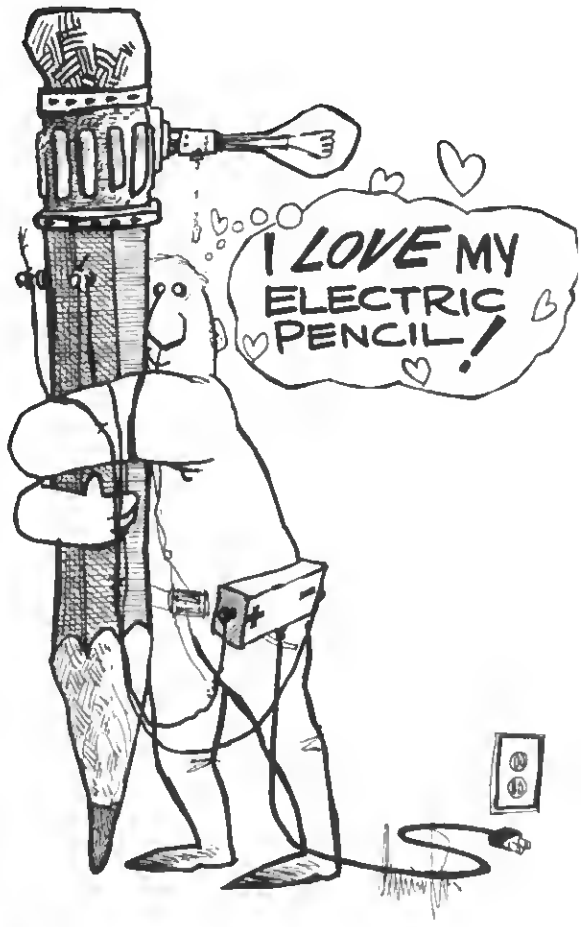
the electric pencil II™

for the TRS-80 Model II* Computer

The Electric Pencil is a Character Oriented Word Processing System. This means that text is entered as a continuous string of characters and is manipulated as such. This allows the user enormous freedom and ease in the movement and handling of text. Since lines are not delineated, any number of characters, words, lines or paragraphs may be inserted or deleted anywhere in the text. The entirety of the text shifts and opens up or closes as needed in full view of the user. Carriage returns as well as word hyphenation are not required since each line of text is formatted automatically.

As text is typed and the end of a screen line is reached, a partially completed word is shifted to the beginning of the following line. Whenever text is inserted or deleted, existing text is pushed down or pulled up in a wrap around fashion. Everything appears on the video display screen as it occurs thereby eliminating any guesswork. Text may be reviewed at will by variable speed or page-at-a-time scrolling both in the forward and reverse directions. By using the search or the search and replace function, any string of characters may be located and/or replaced with any other string of characters as desired. Specific sets of characters within encoded strings may also be located.

When text is printed, The Electric Pencil automatically inserts carriage returns where they are needed. Numerous combinations of Line Length, Page Length, Character Spacing, Line Spacing and Page Spacing allow for any form to be handled. Right justification gives right-hand margins that are even. Pages may be numbered as well as titled.



the electric pencil

–a Proven Word Processing System

The TRSDOS versions of The Electric Pencil II are our best ever! You can now type as fast as you like without losing any characters. New TRSDOS features include word left, word right, word delete, bottom of page numbering as well as extended cursor controls for greater user flexibility. BASIC files may also be written and simply edited without additional software.

Our CP/M versions are the same as we have been distributing for several years and allow the CP/M user to edit CP/M files with the addition of our CONVERT utility for an additional $35.00. CONVERT is not required if only quick and easy word processing is required. A keyboard buffer permits fast typing without character loss.

	CP/M	TRSDOS
Serial Diablo, NEC, Qume	$ 300.00	$ 350.00
All other printers	$ 275.00	$ 325.00

The Electric Pencil I is still available for TRS-80 Model I users. Although not as sophisticated as Electric Pencil II, it is still an extremely easy to use and powerful word processing system. The software has been designed to be used with both Level I (16K system) and Level II models of the TRS-80. Two versions, one for use with cassette, and one for use with disk, are available on cassette. The TRS-80 disk version is easily transferred to disk and is fully interactive with the READ, WRITE, DIR, and KILL routines of TRSDOS.

TRC	Cassette	$ 100.00
TRD	Disk	$ 150.00

✓255

Features

TRSDOS or CP/M Compatible * Supports Four Disk Drives * Dynamic Print Formatting * Diablo, NEC & Qume Print Packages * Multi-Column Printing * Print Value Chaining * Page-at-a-time Scrolling * Bidirectional Multispeed Scrolling * Subsystem with Print Value Scoreboard * Automatic Word & Record Number Tally * Global Search & Replace * Full Margin Control * End of Page Control * Non Printing Text Commenting * Line & Paragraph Indentation * Centering * Underlining * Boldface

VISA

master charge

*TRS-80 is a registered trade mark of Radio Shack, a division of Tandy Corp.

mss

MICHAEL SHRAYER SOFTWARE, INC.
1198 Los Robles Dr.
Palm Springs, CA. 92262
(714) 323-1400

✓ Reader Service—see page 274

"By adding the table starting address to the ASCII code of the original key character, you can find the memory locations now associated with that key."

76	L	N	78	4E
77	M	M	77	4D
78	N	B	66	42
79	O	R	82	52
80	P	L	76	4C
81	Q	?	63	3F
82	R	P	80	50
83	S	O	79	4F
84	T	Y	89	59
85	U	G	71	47
86	V	K	75	4B
87	W	,	44	2C
88	X	Q	81	51
89	Y	F	70	46
90	Z	;	59	3B
91-95	none		91-95	3C-3G
96	@ upper case	@	96	60
97	a	a	97	61
98	b	x	120	78
99	c	j	106	6A
100	d	e	101	65
101	e	>	62	3E
102	f	u	117	75
103	g	i	105	69
104	h	d	100	64
105	i	c	99	63
106	j	h	104	68
107	k	t	116	74
108	l	n	110	6E
109	m	m	109	6D
110	n	b	98	62
111	o	r	114	72
112	p	l	108	6C
113	q	/	47	2F
114	r	p	112	70
115	s	o	111	6F
116	t	y	121	79
117	u	g	103	67
118	v	k	107	6B
119	w	<	60	3C
120	x	q	113	71
121	y	f	102	66
122	z	+	43	2B

Table 1.

VR DATA'S DISK DRIVE HEADQUARTERS

DISK HEAD CLEANERS
5-1/4" $14.95
8" 3M CLEANER
$24.95

77 TRACK DRIVES
TRS-80 Model I compatible
$545
NEW LOW PRICE

40 TRACK DRIVES
TRS-80 Model I compatible
$340
NEW LOW PRICE

● ORDER NOW TOLL FREE 1 (800) 345-8102 ● IN PENNSYLVANIA (215) 461-5300 ●

MODEL II
OVERSTOCK SALE
64K
$3400.

6.3 MEG HARD DISK WINCHESTER
Superbrain, Apple, TRS-80
$2445.

SUPERBRAIN™
by INTERTEC
64K $2995.00
...ready with
5-1/4" 200 drives ● in stock

● TRS-80 Disk & Other Mysteries $22.95 ●

MINI DISK DRIVE EXTENDER BOARD
$14.95

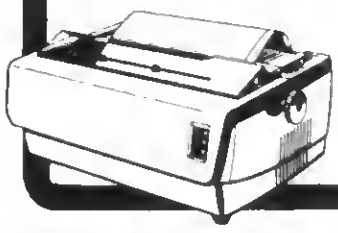
EXTENDED 1 YEAR WARRANTY
MINI DISK DRIVES $45.00
Call For Details

SOFTWARE	MOD. I	MOD. II
Word Processing (Magic Wand)		300
General Ledger	$149.95	249
Payroll	99.95	199
Data Base	149.95	299
	Tape	Disk
Upper/Lower Case Modification	$19.95	$24.95
Comprehensive Diagnostics	34.95	34.95
CP/M	$175.00	
New DOS + 40 TK	100.00	
New DOS/80	145.00	

Software Documentation Available ● CALL FOR PRICES

4K L II TRS-80		575.70
16K L II		789.60
RS-232		92.10
OK Expansion Interface		278.10
16K Expansion Interface		376.10
32K Expansion Interface		474.10
Telephone Modem		179.95
Emulator CRT by Intertec		895.00
CRT Stands	from	139.00
Anti-static Mats		110.00

EPA 250*
*Electronic Printing Ability 250 wpm
DAISY WHEEL PRINTER
by VR Data
$2195.



*Parallel or Serial Interface
*25 cps, 45 cps Optional (Add $600.)
*Tractors Optional (Add $250.)

✓31

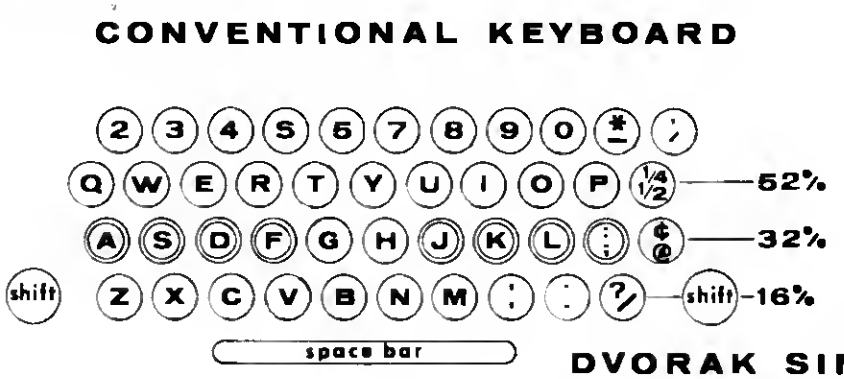
● VISIT OUR NEW WAREHOUSE SHOWROOM AND REPAIR CENTER ●

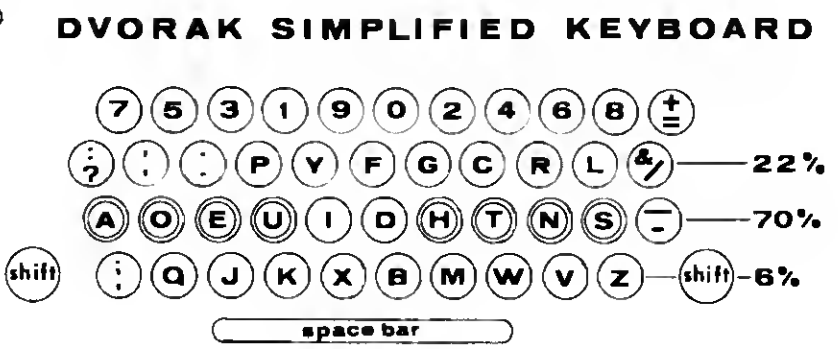
VR Data
777 HENDERSON BLVD.
FOLCROFT, PA 19032

WE SERVICE MANY BRANDS OF COMPUTER EQUIPMENT.
CALL FOR CONSULATION AND ESTIMATE.
DEALER INQUIRES INVITED ● BIDS ACCEPTED ● ABOVE PRICES ARE CASH DISCOUNTED, CALL FOR OTHER TERMS.
ORDER NOW ● TOLL FREE 1 (800) 345-8102 ● IN PENNSYLVANIA (215) 461-5300

✓ Reader Service—see page 274

"The first part of the program initializes the DCB and fetches the original keyboard scanning program address, which becomes the object of a call statement."





Figs. 2a and 2b. Conventionel (QWERTY) end Dvorek Simplified Keyboard (DSK). Relative emount of typing performed on eech horizontel row indicates the superiority of the DSK, with nearly 40 percent improvement et the critical home row.

another eddress into the DCB after turning on the power, another keyboard scanning routine can be substituted for the one in ROM. Instead of writing eh entirely new routine to scan the keyboard, the orlginal ROM is called and modified by this program.

Refer to the essembly languege listing, Progrem Listing 2. The first part of the program initializes the DCB end fetches the original keyboard scanning program address, which becomes the object of a call statement. It then jumps back to the ready messege of BASIC. Jumping to location 1A19H may cause en out-of-memory response, which may be ignored. This return point will also work with Disk BASIC.

BASIC detours through the new program each time the keyboard is scanned. The first thing it does is call the orlginal keyboard driver. If KBFIX or Disk BASIC (which includes keyboard debounce) is used, the address will be different than the 03E3H address of the routine in ROM.

The ASCII code produced for e shift zero is the same as that for the space bar; we need to decode that keypress combination directly. The routine checks for a 01 in keyboard memory locations 3810 and 3880. If both conditions are met, the value of the STAT or status location is toggled from 00 to 80H to 00.

The program then checks the status switch to see if the keyboard charecter should be altered. If the stetus location contains en 80H, the program restores the character in the A register and jumps back to the calling program in BASIC. If the stetus is 00 the cherecter is used to index a character in the lookup table (Table 1). The teble is similer to the one on pages C/1 and C/2 of the Level II BASIC manual.

The conversion table could be shortened by leaving out ASCII codes 0 to 32, all of the lowercase letters, and some of the punctuation cherecters. However, the indexing routine would heve to be more complex to deal with the exceptions. Also, the keyboard is purposely mede es easy es possible for the end user to modify. At current discount prices, the program occupies less than $2.00 worth of memory, so length should not be a major consideration.

To change the character produced by any key on the keyboard it is only necessary to change the data statement associated with that key. The data in lines 179–184 of the BASIC listing corresponds to the characters in the lookup table. It is also possible to change characters on the fly, then POKEing the appropriate ASCII code into the proper memory location. By adding the table starting address to the ASCII code of the originel key character, you can find the memory locations now associeted with thet key.

For example, if the BASIC program told you thet the lookup teble starts at 32600 and you want to change the letter a to m, add 32600 to 65, the ASCII code for e. If you print PEEK (32665), the computer should respond with 65. Then POKE 32665,77, the ASCII code for m and you should now see m displeyed on the screen when you press the e key.

You can also POKE grephic codes 129–191 into the key locations in order to write grephics directly to the screen. Don't be surprised, though, if the graphic blocks look like BASIC keywords when you list a program containing them.

Word Process Problem

A problem arises when you want to use DSK with e machine language program such as Electric Pencil or Scripsit. The conversion program must be patched into the keyboard scanning routines of these programs, since they do not use the keyboard scanning software in BASIC ROM.

To make the keycaps reflect the DSK arrangement, it is easy to pull off the caps with a bent paperclip end put them in the desired order. If you choose to make variants of the punctuetion mark placements found on the classic DSK, it would be eye-pleasing to buy e set of press-on labels the seme diemeter es the keys. Type the exact letters, numerels and punctuation

If you just bought another printer, boy are you gonna be sorry.

Epson.

The Epson MX-80. It's not just another worked-over rehash of last year's model. It's our top-of-the-line 80-column printer. It's new. From the ground up. And it's the most revolutionary printer to hit the market since Epson invented small printers for the 1964 Olympics in Tokyo. Don't take our word for it, though. Compare. There simply isn't a better value in an 80-column printer. Period.

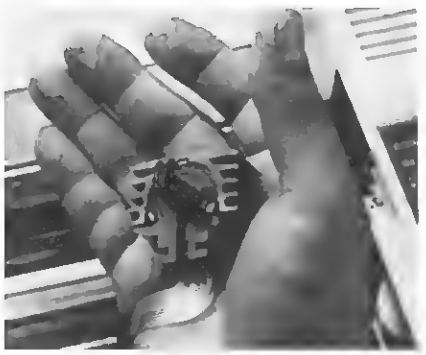
But here's the fact that's going to stand the printer world on its ear. The MX-80 sports the world's first *disposable* print head. After it's printed about 50 million characters, you can throw it away. Because a new one costs less than $30, and the only tool you need to change it is attached to the end of your arm.

Now that's revolutionary, but that's only the beginning. The MX-80 also prints bidirectionally at 80 CPS with a logical seeking function to minimize print head travel time and maximize throughput. It prints 96 ASCII, 64 graphic and eight international characters in a tack-sharp 9x9 matrix. And it provides a user-defined choice of 40, 80, 66 or 132 columns and multiple type fonts.

We spent three long years developing the MX-80 as the first of a revolutionary series of Epson MX Printers. We employed the most advanced automatic assembly and machining techniques in existence to produce a printer that is incredibly versatile, remarkably reliable and extraordinarily inexpensive. It's a printer that could only come from the world's largest manufacturer of print mechanisms: Epson.

If it sounds like we're proud of the MX-80, we are. Not only does it do things some of the world's most expensive printers can't do, it'll do them for you for less than $650. That's right. Under $650.

And if that isn't revolutionary, we don't know what is.



The world's first disposable print head. It has a life expectancy of over 50 million characters, yet it's so simple, you can change it with one hand. And it costs less than – repeat less than – $30.

404

EPSON
EPSON AMERICA, INC.

23844 Hawthorne Boulevard, Torrance, California 90505, Telephone (213) 378-2220

"Type minimum pumpkin in QWERTY; try it quickly . . ."

marks on the labels before removing them from their backing. Draftsman's fixative spray would make the tops of the labels almost as long lasting as the manufacturer's original key caps. It is also a good idea to attach stick-on labels to the key-cap fronts identifying the original QWERTY positions, for use with machine language and other programs not written with the DSK in mind.

One possible modification leaves the number keys in their numerical order. To do this, replace line 178 in the BASIC listing with the following:

```
178 DATA 48,49,50,51,52,53,54,55,56,57,58
```

Also exchange the 33 and 42 in line 177 to exchange the exclamation point and asterisk. The question merk is now a lowercase character, thus meking it easier to abbreviate print in BASIC programs.

Practicable Persuasion

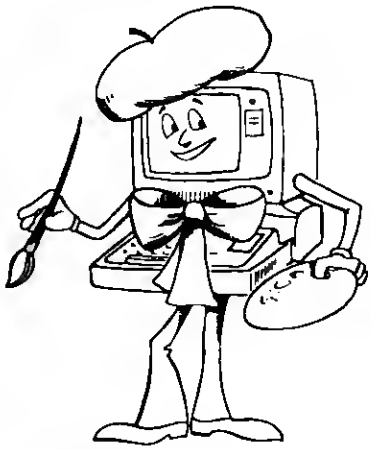
When you have your DSK program debugged and ready for use, consider the practicality of using each arrangement. Start with QWERTY and type federated. Note that all letters are keyed by fingers of the left hand. Now press shift zero and try the same word in DSK. (Refer to the DSK chart, Fig. 2b, but keep the same home-row finger position that you learned for QWERTY. Home-row keys are double-circled on the chart for operators who have not previously learned the touch system of typing.) It will be slow going at first, but note how *f e d e r a t e d* alternates right, left, right, left. This is one of the open secrets of why DSK is highly superior.

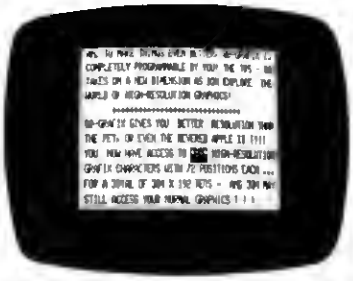
Try December in each mode—note the difference in feel as you type it in DSK. Type minimum pumpkin in QWERTY; try it quickly, with as much speed as you can muster. Now switch to DSK and, referring to the chart, type minimum pumpkin a few times slowly, noting how easily it flows on the new keyboard. After a dozen times, you will by typing as fast as you were QWERTY style.

If you haven't become intrigued by now, try copying some plain text first on QWERTY, then on DSK. You will find yourself remembering certain placements as the e and o without having to refer to the chart. You are learning DSK already! That's how easy it is. ■

FROM PROGRAMMA
HI-RESOLUTION GRAPHICS FOR THE TRS-80®

LOWER CASE
The 80-GRAFIX board includes two sets of lower case characters at no additional cost.





INVERSE VIDEO
The 80-GRAFIX board allows you to do inverse video to high-light your screen displays.

DEMONSTRATION PROGRAMS
The 80-GRAFIX board is supplied with a Character Generator software and several demonstration programs.

FINALLY, AT LAST. . .

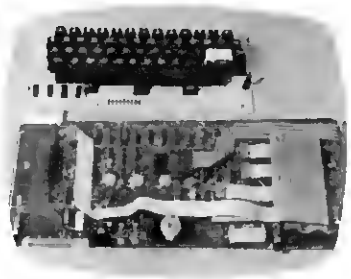
HI- RESOLUTION GRAPHICS is available for your TRS-80 computer system. The 80-GRAFIX board from PROGRAMMA International, Inc. gives your TRS-80 high resolution capability that is greater than the Commodore CBM/PET or even the revered APPLE II.

80-GRAFIX gives the TRS-80 an effective screen of 384X192 pixels, versus the normal 127X192 for the TRS-80, 80X50 for the CBM/PET, or the 280X192 of an APPLE II. As an added feature, 80-GRAFIX offers you lower case characters at no additional cost. Of course, you can also create your own set of up to 64 original characters using the supplied Character Generator software.

The 80-GRAFIX board is simple to install (note that this voids your Radio Shack warranty), and programming is done through BASIC. 80-GRAFIX opens up a whole new realm of software development and excitement never dreamed of for the TRS-80!

CHARACTER GENERATOR
The supplied character generator software allows you to create your own character set of up to 64 original characters.

REAL-TIME GRAPHIC GAMES
With the 80-GRAFIX board you can write exciting real-time games using BASIC.



EASY INSTALLATION
The 80-GRAFIX board is simple to install and fits inside the TRS-80 case.

GRAPHICS GALORE
The 80-GRAFIX board and the supplied Character Generator allow you to become an artist.

ELECTRONIC DESIGN
The 80-GRAFIX board has unlimited application in Electronic design and Education.

80-GRAFIX HI-RESOLUTION
Finally, the only means to protect your computer investment is to order an 80-GRAFIX board TODAY!

EXCITEMENT & FUN
Open up a new realm of software development with the 80-GRAFIX board.

Available exclusively through PROGRAMMA at the cost of $149.95
Please check with us for availability prior to ordering
VISA and MASTERCHARGE accepted
TRS-80 is a registered trademark of the Tandy Corp.

PROGRAMMA INTERNATIONAL, INC.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579 · 384-1116 · 384-1117

The essence of variables.

Into the 80's

Ian R. Sinclair
89 Alexandra Road
Sible Hedingham
Halstead, Essex CO9 3NP
England

Last time, you remember we were faced with the problem of selecting a pair of words at random from our data list, doing it by running through a random number of items and discarding the ones we didn't need. All this effort wes necessary because we couldn't pick a single word at random out of the list. Now we're going to look at a method of doing that.

This method is a dern sight simpler than the name suggests. We READ the data as usual, but as each item is read we label it as a string variable, and number the variables. When we read off the first animal name, we number it as Q$(1), and we make its answer A$(1). Similarly, the next pair become Q$(2) and A$(2), the next are Q$(3), A$(3), etc.

Storing Variables

The computer stores these variables and refuses to be confused by eny similerities between variable names. A$, A1$ and A$(1) are three separate variables which will be stored in different parts of memory and can be called up only if you use the correct titles.

One advantage of storing words like this is that we can retrieve any question or answer pair without having to sort through *all* the words. If our random choice comes up with the number two, we can then print Q$(2), and match the answer at the INPUT stage with A$(2). Remember that RND(6) generates the random numbers.

A set of strings tagged with numbers in this way rejoices in the splendid title of an array of subscripted string variables. Array means a list, and subscripted means that we've tagged each item with numbers so that we can identify them.

My friends, nothing could be easier than setting up an array now that you know about the FOR.....NEXT loop. The arrey in Listing 1 starts with the FOR N = 1 TO 6 statement, which means we start with the value of the variable N set to one. The next command is READ Q$(N), A$(N). This reads in the first word of data and assigns it the variable name Q$(1), because N is set to one. The next word is also read and assigned the variable name A$(1). That's the first question and answer pair dealt with, so the next command is NEXT. This causes the computer to increase the value it has assigned to N, and compare it with the limit we set at the start, which was six. We've just increased N from one to two, so the NEXT command moves to the READ command with N set to two. The next two words are read, assigned the variable names Q$(2) and A$(2), the control returns to the NEXT statement, and N is set e three, and compared with six. This goes on until N hes been set to six. At this value, the last pair of question and answer words are read in and assigned the variable names Q$(6) and A$(6), when N is increased to seven by the NEXT step, and the loop is broken because seven is greater than six. All the data words have been read and converted into an array so we can pick them out as we want, using a piece of program like the one shown in Listing 2.

Dimensioning

Before you start using tagged variables, there's one more instruction you need to know. When you set up an array of tagged variables, the computer stores the variables in one part of its memory and keeps a note of them, elong with the tags in another part. So it can organize this process efficiently, it needs to be told how many tags you might use. Might use, notice, not did use. If you specify that you might use 50 tags, but use only 20, that's all right by the TRS-80, but if you specify that you might use 50 and then try to use 51, you'll get an error message (BS) whenever you try to use the last tag, meaning that you haven't reserved enough memory space.

The number of tegs which you might use on a variable is called the dimension of the variable. If you're going to enter 12 names, assigned to L$ and tagged L$(1),...L$(12), the dimension of L$ is 12. The TRS-80 allows you to use dimensions of up to ten on any subscripted variable without any extra work, but if you are going to use more than this number of tags you have to enter the dimension early in the program, by using the DIM (for DIMension) statement.

DIM L$(12) means that you plan to use a subscripted variable L$ with subscript numbers which do not exceed 12. If your program has several subscripted variables, you don't need to write a separate line of DIM for each. For example, you can write DIM L$(12), P(20). Make sure that you haven't reserved more memory space than your computer has, and make sure that the DIM statement comes early in the program, well before you are going to use any of these subscripted variables. Remember also that you can use 0 as a subscript, so you can have L$(0), L$(1)...which lets you have an extra subscript without having to reserve any more memory.

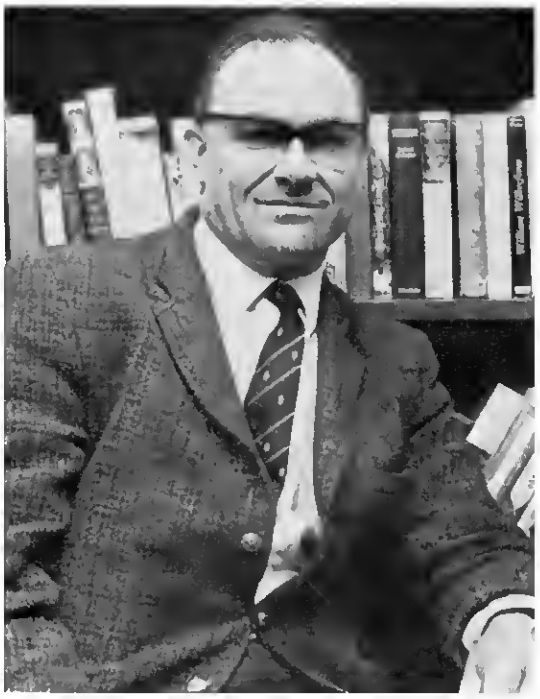
Loading the Strings

Listing 3 allows you enter word pairs

Ian Robertson Sinclair was born in 1932 in Tayport, Scotland, and educated at Madras College, St. Andrews where, needless to say, he played golf. He graduated in 1955 with a B.Sc at the University of St. Andrews. He started writing articles for magazines in 1964 and began teaching college in 1966. His first book, Understanding Electronic Components, *was published in 1972 (still in print) and he is now working on his thirtieth.*

Writes Sinclair, "I bought a TRS-80 whenever a keyboard became available over here (to English TV standards) and have eaten, drunk and slept TRS-80 computing ever since."

I. R. Sinclair



from the keyboard and tag them as subscripted variables. As an added refinement, the tag number is shown alongside the word as you type it in.

This is a powerful piece of program. It starts with FOR N = 1 TO 100. I picked 100 as an upper limit, but we could have used anything else. The point of doing this is to ensure that we don't put in more words than we've allowed for in the DIM statement.

N is set to one at the beginning, and the PRINT N: command prints a one on the screen, followed by the question mark which always goes along with the INPUT statement. The semicolon after PRINT N is there to make sure that the N and the INPUT word are on the same line. You can then type a question word, which will be assigned the variable name of Q$(1). Follow this with a comma, and type the answer word, A$(1) and ENTER. What happens if you forget the comma and then use ENTER? Disaster, because the two words will simply be strung together as a single word assigned as Q$(1), and the computer will print two query marks to tell you that it's waiting for the answer word. The TRS-80 is a great little computer, but it can't correct your mistakes. If you enter each word separately, you'll get a single query mark as a prompt for the question word, and a double query mark as the prompt for the answer word.

The next section of the line is an IF... NEXT...ELSE decision. If Q$(n) is not assigned to X, then the next value of N is taken and another pair of words can be used. If the letter X is input, the program breaks out of the FOR...NEXT loop, and makes N take a value one less than that assigned to it last. In this way you avoid using the value X in the program. If, for example, X is the tenth item which you entered, then N is reduced from ten to nine, because there are only nine actual items. We can now pick out any pair of words for our program and enter them from the keyboard. The idea of subscripted string variables (you can have subscripted number variables as well, but strings are more fun) is useful, but it can be extended.

A Matrix

Take a look at the program in Listing 4. It starts at line 10 with something which is new to you, a pair of FOR-NEXT loops together, one inside the other. This is called nested, because the first one completely surrounds the second one and the second the third (if you have one) and so on. There are two here, and they are reading what looks at first sight to be a single string variable, A$. A$, however, is a subscripted string variable, and it's not singly subscripted like A$(1), but doubly subscripted like A$(1,1). This arrangement of subscripts is set up by the use of I and J in the FOR-NEXT loops and makes for very economical programming, because in place of question string and answer string we simply have A$. It's important but difficult to grasp if you've never used anything like it before, so we'll spend some time looking at this one closely.

The first, or outside, loop starts with FOR I = 1 TO 4, so that on the first run I is given the value one. The program then moves to the second FOR instruction, and sets J at one. The READ instruction causes the first word of data to be read and labeled as A$(1,1) because I = 1 and J = 1. We would normally have two separate NEXT statements, but in this type of array we can get away with the one which is shown, NEXT J,I, which means take the next J if there is one, and if there isn't, take the next I.

Notice that you have to be fussy about the order of these variables. The NEXT variables have to be in reverse order from the FOR variables, so that if the first FOR uses I, then I must be the last variable in the NEXT. If you don't do this, your nest has holes in it. For example, if we opened with FOR X = 1 to 5: FOR Y = 1 to 4: FOR Z = 1 to 2, we would have to finish with NEXT Z,Y,X.

So far we have read the data word HORSE and assigned it as A$(1,1). We then take the next J, keeping I at one, and so making J = 2. The next word, FOAL, is assigned the string coding A$(1,2). Starting to look interesting?

We're out of J's now, so the next I is taken, and I now has the value of two. This time around, with I = 2 and J = 1 (because we started back at the FOR J = 1 to 2 again), we'll read PIG and assign it as A$(2,1). The inner loop will then cause PIGLET to be read, and assigned as A$(2,2). In fact we're assigning four sets of two words when the program has run. If you like a more abstract description, it's four lines of animals with two columns, one for parents, the other for the young. Mathematicians (may they be preserved...preferably in aspic) call this arrangement a matrix.

The nicest thing about a matrix of this sort is that it's easy to make neat arrangements. Line 30 gives you some idea of what can be done. Starting with the FOR statements which set up the matrix arrangement, it uses J in a PRINTTAB( ) statement to space the two columns of words neatly on the video screen. The semicolon after the A$(I,J), makes sure that the young animal's name get printed in the same line as the old one. Follow this up with a separate PRINT command between the NEXT J and the NEXT I, or the computer will try to print everything on the same line, and fail miserably. It seems a shame to abandon that NEXT J,I already, but the results are quite satisfying. Run it and see!

Cutting Strings

And now, as they say, for something different. Remember, a month or so ago, when

```
500 FOR N=1TO6:READ Q$(N),A$(N):NEXT
510 FOR N=1 TO 6:PRINT Q$(N),A$(N):NEXT
```

Program Listing 1

```
150 DATA"LION","PRIDE","WHALE","SCHOOL","FISH","SHOAL",
    "SHEEP","FLOCK","COWS","HERD","GEESE","GAGGLE"
499 REM INTO80'S FIG.4.2
500 FOR N=1TO6:READ Q$(N),A$(N):NEXT
510 R=RND(6):PRINT Q$(R)
```

Program Listing 2

```
10 DIM Q$(100),A$(100)
20 FOR N=1TO100:PRINT N".";:INPUT Q$(N),A$(N):IF Q$(N)<
   >"X" THEN NEXT ELSE N=N-1
30 FOR Y=1TO N:PRINT Q$(Y),A$(Y):NEXT
```

Program Listing 3

```
10 DIM A$(6,6):FOR I=1TO4:FOR J=1TO2:READ A$(I,J):NEXT
   J,I
20 DATA "HORSE","FOAL","PIG","PIGLET","DOG","PUPPY","CO
   W","CALF"
30 FOR I=1TO4:FOR J=1TO2:PRINTTAB(20*J)A$(I,J);:NEXT J:
   PRINT:NEXT I
```

Program Listing 4

```
10 DIM L$(50):FOR N=1TO5:INPUT L$(N):NEXT
100 REM INTO 80'S FIG 4.5 FAULTY EXAMPLE
110 FOR N=1TO50:IF LEFT$(L$(N),1)<>"D"THEN NEXT
120 PRINT L$(N):NEXT
```

Program Listing 5

```
10 DIM L$(51):FOR N=1TO5:INPUT L$(N):NEXT
100 REM INTO 80'S FIG 4.6
110 FOR N=1TO50:IF LEFT$(L$(N),1)="D" THEN PRINT L$(N):
    NEXT:ELSE NEXT
```

Program Listing 6

```
10 DIM A$(51):FOR N=1TO50:READ A$(N):NEXT
20 INPUT "SURNAME"; X$
30 L=LEN(X$)
40 FOR N=1TO50:IF X$<>RIGHT$(A$(N),L) THEN NEXT ELSE PR
   INT VAL(A$(N))
50 DATA "2174678003JOHN DOE","2170322104TIM BUCK" :REM Y
   OU NEED A TOTAL OF FIFTY ENTRIES!
```

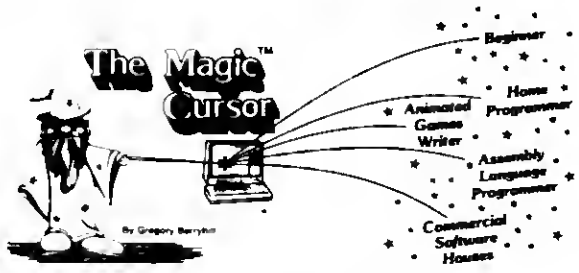
Program Listing 7

learning to recognize a string, using instructions like IF A$ = P$ THEN...? One of the hazards of that type of recognition is that if you print one of these string variable words with a space or a misspelling, the computer simply won't recognize it. We're now going to look at ways around that problem, making use of three very powerful string selection instructions, LEFT$, RIGHT$, and MID$. Let's take 'em slowly, one by one.

LEFT$, as its name suggests, selects the left part of a string. You have to specify

TRS-80* OWNERS:

- **Let the computer write your "Basic" program for you!**
- **Draw pictures, animated figures, data forms!**
- **Create a library of display formats!**
- **Produce "Commercial" grade software!**



The **Magic Cursor** is a Revolutionary Family of Products which provides a dramatic new method of reproducing drawings and displays that you create on your screen. It makes both simple displays and complex interactive data input forms. It stores a "BASIC PROGRAM" on disk (or tape) ready for you to execute alone or as a subroutine. It produces screens in both standard or wide screen.

It is available for any level 2, 16K or larger system with tape or disk. An optional version is now available which creates an assembly language program.

Be sure to pick out the system that fits your present needs and order it today. You may upgrade your original copy by paying the difference and a moderate service charge.

MAGIC CURSOR PROGRAMS

THE MAGIC CURSOR allows you to easily create screens (including graphics) on your video. A powerful command then generates the BASIC instructions to recreate the screen. For the first time, a program for automatic generation of video display forms. (16K Tape or Disk) **$24.95***

THE MAGIC CURSOR I additionally makes sophisticated Data Entry and Display easy. With Magic Cursor I you define the Data Entry or Display fields directly on your screen. The definition commands generate the BASIC instructions to implement the Data Entry and Display. The Magic Cursor I has commands which move, center, and duplicate blocks of graphical or alpha/numeric displays. You can even justify text. (16K Tape Only) **$79.95***

THE MAGIC CURSOR II adds the power to write animated games easily in BASIC. The Magic Cursor II allows you to reload previous screens either from memory or from Disk. You can then modify them and store either the modified screen or only the changes. (32K Disk Only) **$99.95***

THE MAGIC CURSOR III will be available soon for the new Model II Computer (32K One or more Disk) **$149.95***

THE MAGIC CURSOR IV provides the features of Magic Cursor II but stores an assembly language program. (32K Disk Only) **$99.95***

WRITE FOR OUR COMPLETE SOFTWARE CATALOG!!

CUSTOM COMPUTER CENTER, INC.™

For ordering or information write
P.O. Box 58042 Houston, Texas 77058
Attn: Jim Martens
or call: (713) 474-2428

*Please add $1.50 handling charge (Texas residents add 6% sales tax) on mail orders.

NEW RELEASES FROM CCC!!!

A Monitor/Trace program with versions for Model I, Model II and Model III.

Trace-80™ ★

TRACE-80 lets you observe the inner working of a machine language program. It allows you to run a machine language program in slow motion and watch the screen. You can stop execution at any time and examine the current instruction mnemonic and all register contents. You can execute your program and watch each instruction mnemonic and register contents list to the screen in place of normal screen display.

If you have a printer, **TRACE-80** allows you to execute your program in slow motion and watch the screen while your printer simultaneously prints the machine code being executed, the memory location and the instruction mnemonic along with the current register contents.

You can execute a machine language program in slow motion, freeze the action, examine and/or change memory, examine and/or change register contents and then continue the slow motion. You can speed up past common routines and slow down to examine other routines in detail or operate in single step mode.

TRACE-80 allows you to trace ROM as well as RAM because instructions are emulated in a special execution buffer.

FEATURES:

- ★ For both beginner and advanced programmer.
- ★ More than 20 commands.
- ★ Trace-80 is written in machine language.
- ★ Traces both ROM and RAM.
- ★ Level II or Mod I Disk operation.
- ★ Model II Disk version available.
- ★ Optionally prints only "Transfer and Control" instructions.
- ★ Full speed, slow speed or freeze execution modes.
- ★ Memory can be displayed/modified.
- ★ Register contents can be displayed/modified.
- ★ Hex, ASCII and mnemonic display modes.
- ★ Abreviated or full printer format.
- ★ Serial printer output if desired.
- ★ Option of normal screen display, memory display, trace display or clear screen.
- ★ Learn assembly language programming as well as machine coding by watching actual code execution and see assembly language mnemonic.

PARTIAL LIST OF COMMANDS: Load disk file, Trace, Slow Motion Execution, Full Speed Execution, Freeze Action, Single Instruction Execution, Examine and/or Display memory, Examine and/or Display Register Contents, Enable/Disable Screen, Eneble/Diseble Printer, Ascii or Hex Display, Full Screen Memory Display, Line Printer Commands, etc.

TRACE-80/MOD-I & MOD-III (for Level II or DOS) ..**$29.95***
Supplied on tape with 3 versions (16K, 32K or 48K) and complete manual.
TRACE-80/MOD-II **$49.95***

The Restauranteur's Consultant

This food and beverage management tool dramatically reduces the human factors in food cost analysis. Those tasks required to effectively operate any restaurant or food service business. Over a half dozen reports give uniform, accurate and up to the minute information for profitability. And, handling of daily cost changes requires only minutes per week instead of hours, because the Consultant makes all the necessary conversions from your case prices. Reports include

1. Menu Recipes
2. Ingredient Listing.
3. Supplier Master File
4. Complete Listing of Food Cost Analysis.
5. Food Cost Summary
6. Input Data Sheets
7. Batch Update.

Mod I or Mod III, 32K, 2 Disk, Printer or Mod II **$750.00***

Call 713-474-2484 or order by mail. Master Charge, Visa, Certified Check or Money Order accepted. Personal Checks require 14 days to clear. C.O.D. or collect calls not accepted. Software guaranteed for replacement only. Prices subject to change without notice. Some programs supplied on cassette tape. For disk versions the cassette supplied will automatically create a disk file.

*Trademark of Radio Shack, a Tandy Co.

which string you want a chunk selected from, and how many letters you want to take. For example, suppose we have the instruction LEFT$ (A$,3). Whatever word is used as A$, the instruction will select the first three letters on the left hand side. If A$ = "HORSE", then LEFT$ (A$,3) gives HOR. A$ is not affected by this, it is still HORSE. If you had spelled it as HORRES, the computer won't care if it has been instructed to look only at the first three letters. RIGHT$ does the same sort of thing. Suppose we have the instruction RIGHT$(A$,3) and A$ = RABBIT". This time BIT is selected from the word, and A$ is still RABBIT. LEFT$ and RIGHT$ do not delete letters from words, they simple select which letters can be used for other purposes.

LEFT$ and RIGHT$ are useful weapons, but MID$ is a real missile. To use MID$, specify what string you want to operate on, at which letter you want to start, and how many letters you want to select. Suppose we take MID$(A$,2,3). If A$ = ANTELOPE the value of the MID$(A$2,3) is NTE—the selection starts with the second letter and takes in three letters.

Suppose we have a list of names stored as subscripted string variables, L$(N), which means L$(1), L$(2), L$(3) and so on. How many of these names start with the letter D? No, don't sit there and count them, write a program! Something along the lines of Listing 5 might suit us very well, assuming we've used a program to read in 50 names (is your telephone index up to date?). For each value of N, the string name has its first letter compared to D. If the string doesn't start with a D, the next one is taken, but if it does, line 120 commands a printout of the name before going to the next one. We can have two commands of NEXT. This can get us into trouble if the last name does not start with D, because in line 110, the FOR . . . NEXT loop will end, and line 120 will then print L$(N), which we don't need, and asks for the NEXT again. This could cause an error report (BS), meaning that we have exceeded the dimensions we asked for.

Listing 6 shows a neater and flawless method of sorting out these D's. The IF statement sorts out the D's and prints the string, and the ELSE causes the NEXT N to be selected if there isn't a D around. The dimension is chosen to allow the NEXT command to take N to 51 without causing an error message. Now how about selecting all the phone numbers which have the same area code? Let's suppose we have 50 numbers stored in an array K(N). Not K$? Tough luck, you can't do it. All these string commands operate only on strings, not on numbers, which is why so many programs store numbers in string form by simply entering them as strings. STR$ converts any number variable into a string variable. For example, if we have the statement K$ just as thoroughly as if we had written K$ = "234" in the first place. If we have 50 number variables, K(1) through K(50), you just add the line:

```
FOR N = 1 TO 50: K$(N) = STR$(K(N)).
```

Watch the double set of brackets, because if you miss one, you'll get the SN error message. Now that you've got your phone numbers in the form of a subscripted string variable array (gives you a feeling of power just to say it!) you can pick off the area codes by using

```
A$ = LEFT$ (K$(N),3)
```

Once again, we have brackets within brackets, and you have to be sure that you've included all of the brackets.

"As it happens, you often want to do things with numbers which you can't do with strings. . ."

Packing Strings

Suppose you stored each name and number together in one string as K$(N). Quite a savings in memory is gained by doing this, because there's only one string to store for each name/number, instead of a separate pair of strings or a string and a number. How do we separate them so that we can print out something that looks rather more civilized than JOHN W DOE 2141673802? One neat and simple method makes use of the statement VAL.

VAL means—find the number value inside a string. It's usually used when a number has been converted into a string by using STR$ and you now want to convert back. As it happens, you often want to do things with numbers which you can't do with strings, like multiplication, division and subtraction, for example. Addition is a bit different, and we'll be looking at what happens when we use the + sign on strings in a moment.

Many computers use VAL just for converting a number string back to a number, but the TRS-80 BASIC goes one better. If you have a string which starts with a number, like 1024 SUNRISE AVENUE, you can extract the number out of the string by using VAL. If, for example, you run the little program:

```
10 A$ = "1024 SUNRISE AVENUE"
20 PRINT VAL(A$)
```

What is printed out is 1024, the number which the VAL statement finds at the start of the string. VAL can only find a number at the start of a string, however. If you have A$ = JOHN DOE 2174267803, then VAL(A$) is zero, because the number follows the letters.

This doesn't prevent you from writing your own routine, using MID$(A$,N,1) to strip characters off the string one by one and test their ASCII codes to find if they are numbers. The ASCII codes for numbers are 48 through 57, so you could detect numbers anywhere in the string and print them out.

To spearate numbers from names by using VAL, we have to place the number first, coding our number/name in the form of 2172677803JOHN DOE.

Listing 7 essumes that you have a set of data lines which contain your telephone number and name strings. Line 10 is straightforwerd—we are just reading each item end labeling it es a string array A$(N), allowing for 50 items. If you don't want to try 50 for starters, make it two and use just the data in line 50.

Line 20 asks for the surname of the person whose number you want typed. From what you know of computers by now, you should not be surprised to learn that your typing of DOE had better match exactly with the DOE which you have stored in the data line!

In line 30, L = LEN(X), X$ is the string variable assigned to the name you typed at the INPUT stage, and LEN means, measure how many characters are in this string. The answer here is three. If we know how many characters are in the surname, and that the surname is at the right hand side of the string, we can pick the surname out of A$ in line 40, by setting a variable L equal to the length of the name string X, and then find RIGHT$(A$(N),L). We could equally easily have saved a line by writing in line 40:

```
RIGHT$(A$(N),LEN(X$))
```

making sure not to leave out any of the brackets. It's a useful feature of our BASIC that we can use expressions like LEN(X$), as well as simple numbers and variables in

✓ Reader Service—see page 274

```
10 INPUT "NUMBER BETWEEN 1 AND 25, PLEASE";N$
20 N$="0"+N$
30 N$=RIGHT$(N$,2)
40 PRINT N$:GOTO10
```

Program Listing 8

```
10 IF LEFT$(A$,2)=LEFT$(L$(N),2) OR RIGHT$(A$,2)=RIGHT$
     (L$(N),2) THEN PRINT "CORRECT, WELL DONE!"
```

Program Listing 9

```
10 CLS:PRINTTAB(23)"THIS IS THE TITLE"
20 PRINTTAB(23)"==== == === ====="
```

Program Listing 10

```
10 CLS:PRINTTAB(21)"THIS IS ANOTHER TITLE"
20 PRINT TAB(21) STRING$(21,42)
```

Program Listing 11

```
10 CLS:PRINT:PRINT
20 PRINT CHR$(23)TAB(12)"TITLE"
30 PRINTTAB(12)STRING$(5,42)
40 FOR N=1TO1000:NEXT
50 PRINT CHR$(28):PRINT@304,"NEXT LINE OF MESSAGE"
```

Program Listing 12

```
10 POKE 16445,0
20 PRINT"HAPPY BIRTHDAY!":FOR N=1TO1000:NEXT
30 POKE 16445,0
40 PRINT "TO YOU...."
```

Program Listing 13

the RIGHT$, LEFT$, MID$ and other expressions.

In line 40 each item of data is examined, and the correct number of letters on the right hand side is stripped off to compare with X$, which might be DOE. If the last three letters are not DOE, then the next string is taken, and if the last three match up, the final part of line 40 instructs the computer to print the telephone number by taking VAL(A$(N).

Here's another use for RIGHT$. Suppose you have a set of numbers which lie between one and 25, and you want to put them into string form so that each has two digits, like 21, 01, 18, 06... If you write numbers in this way, you can put them into e string and get them back easily, because you always want the same number of characters back, two in this example. If you had these at the end of a string, you could use RIGHT$ (A$(N),2), for example.

Listing 8 shows how this operation of padding numbers out can be achieved. The number in this example is typed in as an answer to the INPUT query, and we take the chance to assign it to a string variable, N$. In line 20, we redefine N$ as being equal to one space plus the old value of N$. When we use a + sign with two string quantities, the quantities are simply run together, or concatenated. If we had typed N$ = "*" + N$, then with N$ = "2", the result would be *2. As it is, line 20 uses a zero between quote marks so that the new N$ consists of the number we read in with a zero in front of it.

In line 30, we define another N$, this time the RIGHT$ of the N$ with a zero in front. If that N$ were 02, the RIGHT$ will give 02, but if N$ were 021, RIGHT$ would give just 21. Either way, the number consists of just two

PROFESSIONAL

HALF A MILLION TAX RETURNS CAN'T BE WRONG!
(OR THEY HAD BETTER NOT BE)

INCOME TAX SYSTEM
FOR TRS-80* MODEL I OR II

Our system, which prepared 500,000 1979 returns, features the following:

1. Full interactive user control, **in tax-form language only,** line-by-line.
2. Screen display of full 1040 and all schedules, prior to printout.
3. Change of a single amount item automatically changes and re-computes entire return.
4. All printout formats IRS and state approved.
5. Stores Preparer's Identification for automatic printing at bottom of page 2.
6. Built-in Validation Check tests entire system, hardware and software.
7. Special Printer Adjustment routines, Line Length, etc.
8. Selection of closed or open output formats—for standard Form 1040 or open name-box types.
9. **Software control of text position on page.** Makes forms-alignment simple. Permits use with non-adjustable printers.
10. Fills in pre-printed Forms or you can use overlays. Your choice.
11. Automatically computes: Tax - SDI Overpayment - Wages Total from W-2's - Earned Income Credit - Income Averaging - Maximum/Minimum Tax - Least Tax Method - All Percentage of Income Limitations - All Fixed Limitations - many, many more.
12. Full support through the tax season — no charge.
13. Inexpensive yearly updates in accordance with tax-law changes.
14. Modular construction — lets you order only the type and size system you need.

PRICING STARTS AT $189.95 (1040 & SCHEDULE A)

25-PAGE DESCRIPTIVE MANUAL $7.50 (Refunded on Order)

MINIMUM SYSTEM REQUIRED: MODEL I, 32K, 1 DISK DRIVE

*TRS-80 IS A TRADEMARK OF TANDY CORP.

CONTRACT SERVICES ASSOCIATES ✓10

706 SOUTH EUCLID **ANAHEIM, CA 92802**

TELEPHONE (714) 635-4055

★ ★ ★ 20 YEARS OF SERVICE ★ ★ ★

characters, and can be selected again by picking off two characters from the string we put it in. Both words and numbers can be padded out in this way to a standard size (two characters, 10, 20, whatever you like so long as it doesn't exceed 255) so that they can be easily selected again.

One small problem arises here. If you have converted a number into a string by using STR$(number), the computer will automatically put a space in front of the number to make room for a negative sign if one is needed. That way, if you use STR$(5), you get a string which is two characters long, and STR$(50) is three characters long, though STR$(−50) is also three characters long. If this might cause problems, one way out of it is to use RIGHT$. To pad out to two characters we use:

```
A$(N) = RIGHT$(" " + A$(N),2)
```

Your numbers will be two-character strings no matter what STR$ has done to them, but watch out for negative numbers!

Time to leave the LEFT, RIGHT, MID business, and look at other things, but before we do, look at Listing 9.

This is one answer to the word recognition part of a mailing list program. If you have the first two letters correct, the comparison is good enough. You have to use this type of recognition carefully, however, because if you have two names which start with the same letters, like ANT or ANTELOPE, the computer doesn't know the difference. In fact, the monkey, donkey problem is the worst you're likely to get!

Presentation

How do you underline a word you have printed on the screen?

There's no way of underlining on the same line as the letters of the word, but if there is space on the next line (*make space*) the problem has a solution. Listing 10 shows one solution—the title words are printed, and, on the next line, using quotation marks, type the characters of underlining. The equality sign and the asterisk are useful for this job. Big BASIC, however, offers a lazy way of underlining in the form of the STRING$ function. STRING$ is a statement which instructs the computer to print identical characters.

There are two ways of specifying which characters we want string together. If we type in:

```
PRINT STRING$(24,"=")
```

the computer will print a string of 24 equality sign. Similarly, PRINTTAB(20)STRING$(24,"A") will produce a row of 24 A's starting at tabulator position 20. Another way makes use of the ASCII codes for the numbers, letters and characters.

"Who is this ASCII, you ask me? It stands for American Standard Code for Information Interchanges, and it's a number-code method of transmitting characters.

Who is this ASCII, you ask me? It stands for American Standard Code for Information Interchange, and it's a number-code method of transmitting characters.

How do you find the ASCII code for a character? The hard way is to look it up in the TRS-80 manual. The easy way is to ask the computer. PRINT ASC("*") will bring up the code which represents the asterisk. We can easily find other codes.

The character which is represented by ASCII code 128 is a blank. It's not the same blank as the one which is represented by ASCII 32. It's possible to have two different blanks. If that sounds weird to you, think of this. The blank represented by 32 can be entered from the keyboard (by using the space bar), but 128 can't. When the computer finds 128 in a string, it can be instructed that this is the end of a string and the start of another. It's a useful distinction.

We can now redesign our underlining statement in line 20, using STRING$, so it looks like Listing 11. There's no reason for these two statements not going into one line, saving memory. Each time you start a new line, you use five bytes of memory, so it pays well to pack the lines as much as possible this way.

CHR$ stands for the character or action represented by the number in brackets following CHR$. For example, PRINT CHR$(68) causes a D to be printed because 68 is the ASCII code for the letter D. Of course, there's a catch: A lot of ASCII codes don't represent letters. They represent actions, and we can have the computer carry them out by using the PRINT CHR$( ) command.

One pair of codes which are peculiar to the TRS-80 are 23 and 28. PRINT CHR$(23) causes the display to print double-size letters and numbers until the command is cancelled by one of a variety of methods. After CHR$(23) has been printed, we have to be careful how we use TAB and PRINT@ numbers because with double-size characters, there are only 32 characters per line.

A command to PRINTTAB(35) isn't going to produce a letter in the middle of the screen. In the same way, the PRINT@ instructions go only to 256, not to 1023.

Double-sized lettering is excellent for titles and for drawing attention to error messages, but the uses suggested in the manual are limited. The character size returns to normal when the CLEAR key is pressed, or CLS used.

Sometimes you don't want to lose what has been printed in the large characters, yet you want more lettering on the screen in smaller print. You can't have a mixture of large and small letters. The PRINT CHR$(23) command operates on the part of the memory which stores the video display characters, and affects either none of it, or all of it at once. One method I use in my own programs is shown in Listing 12.

Line ten skips the first line of the screen. It could be done just as easily by using a PRINT@ command in line 20, but we're opted to use TAB. The CHR$(23) sets up the big letters, and we print the title and underline it. Line 40 simply arranges a time delay so we can sit back and in line 50, the PRINT CHR$(28) then restores the lettering to normal size. It was not intended as a way of restoring normal size letters but as a method of wiping out the top line! The top line was left blank as it will otherwise be wiped clean by the PRINT CHR$(28) command.

Notice also that the lettering which was printed double-size is now normal size but double spaced. It still looks good as a title.

We've also had to position the next line using PRINT@ to keep out of the way and avoid wiping out any other lines.

Another way of getting double-spaced print and returning again uses the instruction POKE. The method is shown in Listing 13. We'll deal with the POKE instruction later; all we need to know for now is that it can change the contents of the memory directly, and more quickly than the usual BASIC instructions. You can mix these commands, using PRINT CHR$(23) to start the big print, and POKE 16445,0 to stop it. For some curious reason, however, the stop command does not work in every program. I have one program in which POKE 16445,0 works perfectly, and another in which it has no effect. I still haven't discovered why.

The sharp-eyed folks will already have sensed that there's more to tell about the PRINT CHR$( ) instruction. It's not particularly useful for printing letters or even punc-

WE WILL NOT BE UNDERSOLD

DISK DRIVES $314

40 track, 102K Bytes. Includes power supply and TRS-80* compatible silver enclosure. Ready to plug-in and run the moment you receive it. Can be intermixed with each other and Radio Shack drive on same cable. 90 day warranty. One year on power supply. Available for 220 Vac (50 Hz) operation. External card edge included.

FDR TRS-80*

CCI-100	5¼", 40 Track (102K Bytes) for Model I	$314
CCI-280	5¼", 80 Track (204K Bytes) for Model I	$449
CCI-800	8" Drive for Model II (½ Meg Bytes)	$795
For Zenith Z89		
CCI-189	5¼", 40 Track (102K Bytes) add-on drive	$394
Z-87	Dual 5¼" add-on drive system	$995

DISKETTES — Box of 10 (5¼") - with plastic library case

Maxell $30 BASF or Verbatim $24

8" double density for Model II (box of 10) $36

CLEAR PLASTIC CASE - Holds 50 diskettes $19

DISK OPERATING SYSTEMS

PATCHPAK #4 by Percom Data $ 8.95

CP/M* for Model I, Zenith $145 • for Model II, Altos $169.00

NEWDOS Plus 40track $ 95.00

NEWDOS 80 $135.00

COMPLETE SYSTEMS

ALTOS 64K, DD, SS, 2-Drive, 1MB ACS 8000-2 $3395

APPLE 16K $888

TRS-80* Model II-64K $3499

TRS-80* LEVEL II-16K with keypad $869

TRS-80* Expansion Interface $249

HEWLETT PACKARD HP-85 $2790

ZENITH Z89, 48K all-in-one computer $2440

ZENITH Z19 $735

TELEVIDEO 912B $698 912C $698 920B $748 920C $748

ATARI 400 $489 ATARI 800 $747

APF Game Only $99 Complete System $499

MATTEL INTELLIVISION $229

MONITORS

LEEDEX	12" B&W Video 100	$129
ZENITH	13" Color	$379
SANYO	9" B&W VM4509	$155
SANYO	12" B&W DM5012	$210
SANYO	12" Green Screen DM5112	$215
SANYO	13" Color DMC6013	$375
APF	9" B&W TVM-10	$139

TELECOMMUNICATIONS

CAT MODEM Works same as Radio Shack Telephone Interface II $148

D-CAT HARD WIRED DIRECT MODEM $199

COMMUNICATIONS SOFTWARE

CCI-TELNET VERSION 5: A communication package which enables microcomputer users to communicate both with large mainframes and other microcomputers. Completely CP/M compatible. Multiple communication protocols supported. $149

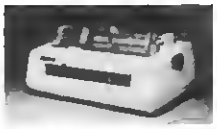
INTELLIGENT TERMINAL SYSTEM ST-80 III: Enables a TRS-80* to act as a dial-up terminal on any time sharing network. $139

16K MEMORY UPGRADE KITS 2 for $70.00 $38

200 ns for TRS-80*, Apple II, (specify): Jumpers $2.50

PRINTERS

NEC Spinwriter



Letter Quality High Speed Printer

Includes TRS-80* interface software, quick change print fonts, 55 cps, bidirectional, high resolution plotting, graphing, proportional spacing: R.O. $2395

R.O. with Tractor Feed $2575 KSR with Tractor Feed $2950

C.ITOH Starwriter, 25 CPS, daisy wheel printer $1895

C.ITOH Starwriter II, 45 CPS, daisy wheel printer $2195

Letter quality printers. Use up to 15" paper. 1 year warranty on parts, 3 months on labor. Proportional spacing and bidirectional printing. Same as VISTA V300.

778 CENTRONICS TRACTOR FEED PRINTER $969

Same as Radio Shack line printer I

737 CENTRONICS FRICTION & PIN FEED PRINTER $795

n x 9 proportional and 7 x 8 mono spacing.

Same as Radio Shack line printer IV

730 CENTRONICS FRICTION & PIN FEED PRINTER $595

7 x 7 matrix Same as Radio Shack line printer II

P1 CENTRONICS PRINTER Same as Radio Shack quick printer $269

PAPER TIGER (IP440) Includes 2K buffer and graphics option $879

(IP460) Bidirectional, 160 cps, graphics and 2K buffer $1075

TI-810 Faster than Radio Shack line printer III. Parallel and serial w/TRS-80* interface software w/u + l case & paper tray $1589

Compressed print, vertical form control $1865

OKIDATA Microline 80 Friction and pin feed $545

Tractor Feed, friction, and pin feed $645

Microline 83 Bidirectional, 120 cps, uses up to 15" paper $1050

EATON LRC 7000 + 64 columns, plain paper $289

ANADEX DP-9500/01 $1350 DP-8000 $795

ACCESSORIES

HEAD CLEANING DISKETTE: Cleans drive Read/Write head in 30 seconds. Specify 5¼" or 8". $20 ea/$45 for 3

FLOPPY SAVER: Protection for center holes of 5¼" floppy disks. Installation tools and rings for 25 diskettes. $ 11.95

Re-orders of rings only $ 6.95

EXTERNAL DATA SEPARATOR: Eliminates data separation problems (crc). Improves reliability. This plug in unit comes fully assembled and tested. $ 29.95

Z-80 SOFTCARD: Your key to software expansion. The plug-in Z-80 Softcard transforms your Apple into a Z-80 while keeping all the benefits of the 6502. Comes with CP/M in two disk format, MBASIC and GBASIC, full documentation and utility programs. $339

RF MODULATOR: Adapts video to TV $ 35.00

TRS-80 & OTHER MYSTERIES $ 18.95

NEC SPINWRITER THIMBLE $11.95 RIBBON $ 6.00

CCS CARDS: Parallel or serial printer interface cards $115.00

RS232: For Radio Shack interface. $ 84.00

TRS232: Teletype current loop output from cassette port $ 49.00

DISK-DRIVE EXTENDER CABLES: Fits all mini-disk drives $ 16.95

SIX (6) PRONG ISOLATOR: ISO-2 $ 54.00

AC FILTER/6 PRONG POWER STRIP $ 39.00

DISK DRIVE CABLES: 2 drive $29.00 4 drive $ 35.00

DUST COVERS: TRS-80/Apple $ 7.95

PLASTIC DISKETTE HOLDER: For ring binder, holds 20 $ 8.00

For fast delivery, send certified checks, money orders or direct bank wire transfers. Personal or company checks require two to three weeks to clear.

DEALER (NATIONAL/INTERNATIONAL) INQUIRIES INVITED **Send for FREE Catalogue**

The CPU SHOP

TO ORDER CALL TOLL FREE 1-800-343-6522

TWX: 710-348-1796 Massachusetts Residents call 617/242-3361

5 Dexter Row, Dept. MC12M
Charlestown, Massachusetts 02129
Hours 10AM-6PM (EST) Mon.-Fri. (Sat. till 5)

Massachusetts Residents add 5% Sales Tax
*TRS-80 is a Tandy Corporation Trademark
• Digital Research

". . . what they don't tell you in the instruction manual is as important and as useful as what they do tell you. . ."

tuation marks. It's just as easy (and easier to follow) if you just type PRINT A or PRINT ; or whatever. CHR$( ) has been found useful for producing effects in a program which we can't get directly from the keyboard, such as the CHR$(23) and CHR$(28). Table 1 shows more of these effects taken from the Level II manual.

There are a lot of ASCII codes which can't be entered from the keyboard but which make their appearance in many programs. These are the graphics characters. One unit of memory, a byte, can store a number of size up to 255; since the highest number of ASCII code for letters or characters is 128, however, that leaves a large number of unused codes. In the TRS-80 these are used for graphics characters. The later Level II manuals include a printout of these characters, but the earlier manuals didn't. For everyone who is now struggling with an old manual, Fig. 1 shows what the graphics characters look like, with their code numbers. To see any of these characters for yourself, look up its code number. Use the command PRINT CHR$(number).

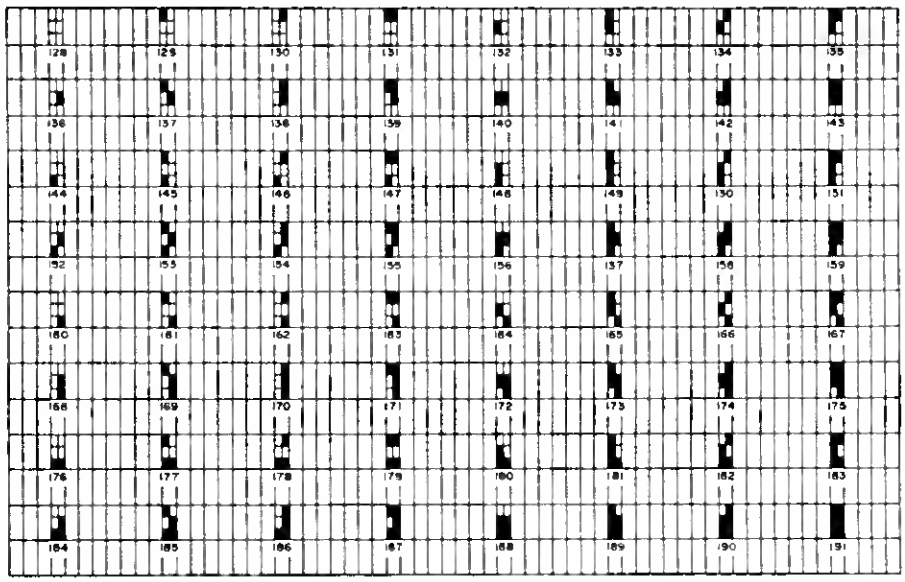


Figure 1

Bigger Graphics

Going onto Sinclair's Second Law - that what they don't tell you in an instruction manual is as important and as useful as what they do tell you - you may have sensed that there's a lot more to this business. If you look at Fig. 2 more is revealed. Each printing position on the video screen consists of six small blocks or cells, and the graphics characters are formed by lighting up various combinations of these cells. Why shouldn't we light up more than one cell at a time in a given block? And there's also no reason why we shouldn't light up more than one block at a time. We can do this by combining codes; Listing 14 shows an example. G$ is defined as the combination of two graphics characters. Each time we command PRINT G$, we'll get that combination, and we can, of course, use the usual printing options of PRINTTAB( ) G$ and PRINT@, G$ to position the set of characters where we want them. We can also use tricks like defining two sets of graphics characters, G$ and H$, and then writing

```
PRINT@N,G$:PRINT@(N + 1),H$
```

which will print the two sets side by side, starting at the position set by the value of the number N (between 0 and 1022).

Alternately we can use:

```
PRINT@N,G$:PRINT@(N + 64),H$
```

which will print G$ at position N, and H$ directly underneath it. Adding 64 to N moves the printing position to the line space immediately below, since we now have 64 print positions in a line. When we used large print we used only 32 characters per line, and there are 64 in the program now.

Next month we'll be looking at the SET and RESET commands, which are a free reign way of creating shapes. Then in the final section of this series we'll investigate the POKE command which can speed up the process of drawing shapes.

Code	Function
0-7	None
8	Backspaces and erases current character
9	None
10-13	Carriage returns
14	Turns on cursor
15	Turns off cursor
16-22	None
23	Converts to 32 character mode
24	Backspace ← Cursor
25	Advance → Cursor
26	Downward ↓ linefeed
27	Upward ↑ linefeed
28	Home, return cursor to display position(0,0)
29	Move cursor to beginning of line
30	Ereses to the end of the line
31	Clear to the end of the frame

Table 1. C/Control, Graphics and ASCII Codes—Control Codes 1#31

INKEY$

INKEY$ can make your program a lot more interesting, it's always by a statement like

```
K$ = INKEY$
```

What is INKEY$? It refers to the value of the character which is fed into the computer when pressing the key just as the computer is scanning the keyboard contacts looking for a key being closed. This scanning takes places continuously when the computer is being used to enter a program, and during much of the time when a program is running in order to detect the BREAK key being pressed. It is halted during a CLOAD or CSAVE, an LLIST or LPRINT. You can't affect what goes on during these operations by punching keys. The RESET button alone, located at the back of the computer, will stop a CLOAD or CSAVE (and will usually corrupt the tape as well). Incidentally, having the continuous keyboard scan means that if you are using a simple keyboard delay routine as a bounce fix, your programs are running slower!

This scan operation is fast but chances

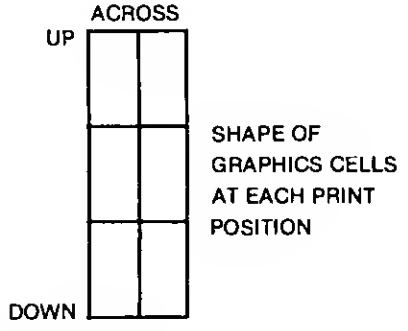


Figure 2

```
10 G$=CHR$(153)+CHR$(166)
20 CLS:PRINT G$
30 PRINTTAB(32)G$
40 PRINT@350,G$
50 END
100 REM TYPE RUN 100 TO RUN THIS ONE
110 G$=CHR$(154)+CHR$(165):H$=CHR$(103)+CHR$(107)
120 CLS:PRINT@150,G$;:PRINT@155,H$
130 FOR N=1TO1000:NEXT
140 CLS:PRINT @400,G$:PRINT@544,H$
```

Program Listing 14

```
5 PRINT "PRESS ANY LETTER OR NUMBER KEY"
10 K$=INKEY$:IF K$="" THEN 10
20 K=VAL(K$): IF K=0 THEN PRINT"YOU ENTERED THE LETTER
    ";K$
30 IF K<>0 THEN PRINT "YOU ENTERED THE NUMBER "; K$
40 END
```

Program Listing 15

```
1000 A$=""
1010 K$=INKEY$:IF K$="" THEN 1010 ELSE PRINT K$;
1020 A$=A$+K$:IF LEN(A$)<2 THEN 1010
1030 IF LEN(A$)=2 AND A$="NO" THEN M=2:GOTO2000
1040 IF LEN(A$)=3 AND A$="YES" THEN M=1:GOTO2000
1050 IF LEN(A$)=2 THEN 1010
1060 IF LEN(A$)>3 OR A$<>"NO" OR A$<>"YES" THEN GOTO201
    0
1070 END
2000 IF M=1 THEN PRINT "  THE ANSWER IS YES":ELSE PRINT
     "  THE ANSWER IS NO"
2005 END
2010 PRINT "  YOU HAVE MADE A MISTAKE- PLEASE TRY AGAIN
    .":FOR N=1TO500:NEXT:GOTO1000
```

Program Listing 16

are, if you just wrote K$ = INKEY$ into a program and let it run, there wouldn't be a key pressed down at the instant when that line of program was carried out, so K$ would be a blank string at first. We get around this by looping around the instruction; forcing it to repeat itself until something is entered by pressing a key. A line such as:

```
50 K$ = INKEY$: IF K$ = "" THEN 50
```

does just that. If the value of K$ is a blank, the line runs again. It keeps running until a value is entered from a key. The value of that key is then stored as K$. If the key is just a letter key or a number key, its value can be printed by making the line read

```
50 K$ = INKEY$: IF K$ = "" THEN 50 ELSE PRINT K$
```

This is a useful way of entering letters or numbers without hitting ENTER. An example of this sort of thing is shown in Listing 15. In line 10, K$ is set equal to INKEY$, and looped back waiting for a key to be pressed, the number value of K$ is found by using K = VAL(K$). If the key is a letter key, its number value is zero, and the message in line 20 is printed. If the key is a number key, its number value is not zero (unless it is the zero key) and the message in line 30 is printed.

Take a look at the program in Listing 16. This is useful for YES/NO choices, because it lets you see the word build up on the screen, and returns at once when the correct word is selected, without needing to hit ENTER. In addition, it signals back to the main program what has been typed, using M = 1 to mean YES and M = 0 to mean NO. Try it out, and then think how it might be improved. Perhaps a flashing asterisk or dash to remind when you need enter another letter?

Next month, we will look at how the TRS-80 makes calculations, formulae and logic easy for us. ■

✓Reader Service—see page 274

You'll be one after reading this introductory lesson on arrays. Comes complete with homework.

A Manipulative Wizard

John D. Adams
13126 Tripoli Ave.
Sylmer, CA 91342

The TRS-80 is a talented little machine. As you learn about it, its possibilities widen surprisingly. And when it comes to handling large data groups with items in special relation to others, it is, indeed, an electronic wizard!

Anyone who has data that must be manipulated should be familiar with the array capability of the computer.

Moving from Level I, where arrays are severely limited, to II is like moving from a Tonka toy to a Mack Truck. The Level II manual has a good section on arrays—if you already know about them. My first attempts to use it were frustrating and confusing because it is somewhat skimpy on details. I learned the hard way—by trial and error.

Arrays

So what is an array? An array is a formal, structured arrangement of information in which individual items of data are related. Gosh, that sounds grim! Is this going to be another one of *those* articles? Nope. Let's take it from the top.

Everyone deals with arrays of one sort or another. Your telephone book is an array. For any page the seventeenth name from the top will correspond with the seventeenth number from the top. A street atlas is a different kind of array. If the map area you are looking for can be found on page 36 with horizontal coordinate E and vertical coordinate 5, then that area has an array location of 36,E,5. Financial reports, bank statements, income tax tables and bills are all arrays. They all have "grouped" information.

How does the computer handle arrays? If you have spent 15 minutes plus with your computer, you know how finicky it is about directions. A quick review of some facts concerning computer memory locations is worthwhile.

The manual explains that a single letter may be used to designate a memory location. With 26 letters in the alphabet, that creates 26 memory locations. Next, the manual states that a single letter and a single digit, such as R4, can be used. With ten digits from 0 to 9, 260 more memory locations can be designated. Finally, two letters, such as EM, can be used to name another 676 locations.

Although this accounts neatly for more than 900 memory locations, it is not quite as generous as it appears. It would be relatively easy to fill all the locations with data, leaving no location in memory for program use. Storing data in this fashion also costs time and space—and locating information can be a nightmare. Think about designing a routine to locate a particular item by scanning all 962 locations beginning with A and ending at Z, then A0 to Z9, and finally AA to ZZ.

Arrays handle data storage by using specialized location names, and make storing, searching and retrieving information almost effortless. They do this using "subscripted" variables.

The character set used in the TRS-80 has no small numbers to use as exponents or subscripts. Exponents are expressed by the up arrow. Subscripts are enclosed in parentheses. For example, if an array is set under the variable M, then M(1) could represent the first item, or "element," in the array, M(27) the 27th, and so forth. (Zero is usable as a subscript and should be used unless you want the location number and the item number to be the same.) A(1), A, A1, A$, A1$ and AA are all different memory locations.

Your data can be stored without touching the standard memory locations. By using a loop, the entire list can be searched quickly for a particular bit of data. When there is more than one type of data to be stored, multi-dimensional arrays or several arrays can be set up so that information in A(5) will correspond to information in B(5), which will correspond to information in C(5), etc.

The simplest type of array, a "one-dimensional" array, is sometimes called a list, because that's exactly what it is. Let's assume you want to store 11 names. On power up the TRS-80 has 11 locations set aside automatically so you do not have to "dimension" the array (which will be discussed later).

There are several ways in which this can be handled. One alternative would be to store each name in a separate memory location. Should the names be alphabetized? Putting things in alphabetical order is a convention designed to make things easier for humans. The computer really doesn't care. It will take the same time to retrieve all 11 names either way. If you do want the names in alphabetical order, the computer will do that for you too. Names loaded in this fashion might look something like this:

```
10 A1$ = "Burke, Samuel"
20 A2$ = "Caldwell, Louise"
 .
 .
 .
90 A9$ = "Smith, Walter"
100 B1$ = "Thomas, Anne"
110 B2$ = "Young, Denise"
```

"Probably the nicest feature of an array is. . . to search out a particular feature without fuss. . ."

Aggravation abounds in this system. Each time you enter a name, you must enter the line number, the variable name, an equals sign and the name in quotes. Doing this with 500 names could cause temporary insanity. Entering 11 names taken at random from the phone book in this fashion used up 304 bytes of RAM, or slightly less than 28 bytes per name. This means that in a 16K machine with 15,527 available bytes, entering 561 names will shoot your RAM. Retrieving a name from this list would be another headache, requiring a comparison between a string and each location.

Now is the time to call the array into function. Loading an array is usually done with a FOR-NEXT loop. Having chosen N$ as the variable name for an array, prepare a loading routine that looks like this:

```
10 CLEAR 150
20 FOR X=0 TO 10
30 INPUT"ENTER NAME";N$(X)
40 NEXT X
```

Line 10 clears enough string space for the names. Allowing an average of 15 or 16 spaces per name should suffice. Be sure to put all CLEAR instructions at the very beginning of your program. If the computer encounters a CLEAR after data has been entered, it will callously throw out your data. Line 20 originates the loop and sets the value in X to zero. Line 30 stops execution so you can enter a name, then stores the name in N$(X). At this point the location is N$(0). Line 40 returns execution to line 20 and increments the value in X by one. This indicates that on the next pass the second name will be stored in N$(1). When the value in X is greater than ten, execution skips to the line following the NEXT instruction. Now then, that's not too difficult, is it?

Enter the lines and run them. They only use 50 bytes of RAM. All you need to do to enter the names is to type them when requested. However, if you enter names with commas in them, the computer misunderstands. It will load what precedes the comma, regard the comma as a data separator and display ?EXTRA IGNORED. This is no big problem: Enter first names first with no commas, or keep the alphabetical order and commas by entering the names within quotes. (When I loaded the same 11 names in this manner, I used only 153 bytes, saving 151 over the original 304.)

Getting the computer to give you back the data is just as simple. Add the following lines:

```
50 FOR X=0 TO 10
60 (L)PRINT N$(X)
70 NEXT X
```

The procedure is just about the same except that line 60 gets data out whereas line 30 puts it in. The (L) in line 60 is for outputting to a printer. If you want a listing on the monitor, omit the (L).

Nicest Feature

Probably the nicest feature of an array is that it enables us to search out a particular item without fuss or bother. Here is a routine for retrieving and printing a particular name; add it to the previous lines.

```
80 INPUT"ENTER NAME TO BE FOUND";S$
90 FOR X=0 TO 10
100 IF S$=N$(X) THEN 130
110 NEXT X
120 PRINT"NAME NOT ON LIST":GOTO 80
130 (L)PRINT N$(X)
```

Line 80 allows you to enter the name you want and put it into S$. Lines 90 through 110 compare it with each name on the list. If the name can't be found, line 120 is printed and you are returned to line 80 for another try. If the name is found, line 100 sends execution to line 130 for printing.

An input loop, an output loop and a search loop make up the skeleton of any array program. Arrays can get a lot more complex, but basically they are all built on this framework.

The three loops used above can be put together in a working program designed to store, print and search an 11-name array. The program is given in Listing 1 with some CLS's and PRINT's to format the material on the monitor screen. Lines 20 and 30 give the user a choice, and some GOTO's were

Store	Jan.	Feb.	March
1	$328.14	$127.40	$552.13
2	87.18	330.12	220.90
3	703.58	941.30	128.58

Table 1.

```
10 CLS:CLEAR 150:PRINT"NAME LIST":PRINT
20 INPUT"DO YOU WANT TO (1) LOAD (2) PRINT OUT (3) SEAR
    CH";Y
30 ON Y GOTO 40,70,95
40 FOR X=0 TO 10
50 INPUT"ENTER NAME";N$(X)
60 NEXT:CLS:GOTO 20
70 CLS:FOR X=0 TO 10
80 PRINT N$(X)
90 NEXT:PRINT:INPUT"PRESS ENTER TO CONTINUE";C$:CLS:GOT
    O 20
95 CLS
100 INPUT"ENTER NAME TO BE FOUND";S$
110 FOR X=0 TO 10
120 IF S$=N$(X) THEN 150
130 NEXT
140 PRINT"NAME NOT ON LIST":GOTO 100
150 PRINT N$(X):GOTO 20
160 REM * END OF LISTING # 1 - NAME LIST *
```

Listing 1. Name List

```
10 CLS:CLEAN 2000:DIM N$(30),A$(30),C$(30),Z$(30),T$(30
    )
20 PRINT"ADDRESS BOOK":PRINT
30 INPUT"DO YOU WANT TO (1) LOAD (2) PRINT OUT (3) SEAR
    CH";Y
40 ON Y GOTO 50,110,140
50 FOR X=0 TO 29
60 INPUT"ENTER NAME";N$(X)
70 INPUT"ENTER ADDRESS";A$(X)
75 INPUT"ENTER CITY AND STATE";C$(X)
80 INPUT"ENTER ZIP CODE";Z$(X)
90 INPUT"ENTER TELEPHONE NUMBER";T$(X)
100 CLS:NEXT:GOTO 30
110 FOR X=0 TO 29
120 LPRINTN$(X):LPRINT A$(X):LPRINT C$(X),Z$(X):LPRINT"
    PHONE  ";T$(X):LPRINT
130 NEXT:GOTO 30
140 INPUT"ENTER NAME TO BE FOUND";S$
150 FOR X=0 TO 29
160 IF S$=N$(X) THEN 190
170 NEXT
180 PRINT"NAME NOT ON LIST":GOTO 140
190 LPRINT N$(X):LPRINT A$(X):LPRINT C$(X),Z$(X):LPRINT
    "PHONE  ";T$(X)
200 GOTO 30
```

Listing 2. Address Book

added to bring you back to convenient places. But look at lines 40 to 60, lines 100 to 150 and lines 70 to 90 and you will see the three basic modules, an input loop, a search loop and an output loop.

What happens if we have more than 11 names, say 50? Try changing lines 40, 70 and 110 to read "FOR X = 0 TO 49". When you run the program the monitor will show ?BS ERROR IN 40. (No—the letters stand for beyond subscript.)

Remember that the TRS-80 sets aside space for an 11-element, or member, array on power up. For larger arrays you must use the DIM(n) statement. To store 50 names, the statement :DIM N$(50) must be added to line 10, as well as changing lines 40, 70 and 110. Otherwise the program remains the same. The DIM(n) statement merely reserves space, or dimensions arrays having more than 11 elements.

This basic program can be enlarged to contain not only names, but addresses, zip codes and telephone numbers. You will still be using one-dimensional arrays, but we will be using five of them: N$ for names, A$ for street numbers, C$ for cities and states, Z$ for zips and T$ for phone numbers. Listing 2, Address Book, shows a program like this that will handle 30 names.

Examine it carefully and compare it with Listing 1. Note the similarity of structure. Isolate the three basic routines. Instead of handling one array, the loops are now handling five. Enter the program and RUN it. Try changing it to search for an address instead, or a phone number or a city and state. Such changes are minor and easy to make.

Multi-Dimensional Arrays

Now we can forge ahead to arrays which are complex—and more useful. Five arrays were used in Address Book: one for each information item. There are good reasons not to simply load all of the information into one array such as N$. First, the strings must be identical for the computer to match them. If all of the information was loaded under N$, the only way for the computer to find an item would be to enter it exactly as it was originally loaded. If you had all that information at hand, you wouldn't need the program. More important, listing the data under one variable name would have seriously hampered the data search. There would have been no way to find an address from a phone number, and no way to find people who live in the same zip code or telephone area. Using the five arrays gives a flexibility in searching techniques, and is pertinent to arrays which have more than one dimension.

To explore the two dimensional array use Table 1.

The money amounts in Table 1 are given in the two categories of store end month. There are consequently three rows and three columns of date. (The store numbers could be considered as another column, if needed.) To find a particular figure use the row and column. For example, the figure for the second store in the third month will be found in the second row and the third column.

Using the figures in the table, you can set up what is called a three by three array. This is a two-dimensional array. Whereas the capacity of a single dimension array is the last number used as a subscript, the capacity of this array will be the product of its dimensions. That is, 3*3 = 9 available locations. Arrays of this type are quickly loaded, printed out and searched by using "nested" loops. The array will be given the name A, S will be used to represent the stores and M to represent the months. Note that these are not string locations (such as A$). Numbers loaded into string locations are regarded as symbols and not as values. Using these variable names, the array will have the name A(S,M). A routine for loading data would be as follows:

```
10 FOR S = 1 TO 3
20 FOR M = 1 TO 3
30 PRINT"ENTER FIGURE FOR STORE";S;"MONTH";M
40 INPUT A(S,M)
50 NEXT M
60 NEXT S
```

There are two loops here, one contained inside, or nested, within the other. Operation of nested loops is not complicated. Lines 10 and 20 originate the loops and set the values in S and M to one. Line 30 asks for information and requests information for the particular store and month represented by S and M. Line 40 deposits that information into location A(S,M) which is presently A(1,1). Line 50 returns execution to line 20 and increments M by one. When M is greater then three, execution skips to line 60. The NEXT instruction sends the computer back to line 10, which increments S by one and starts the nested loop working again.

Do you see that the nested loop (M) has to cycle three times before S is incremented? This produces subscript values (1,1), (1,2) and (1,3). After the value of S is incremented, the interior loop cycles three times again, producing values of (2,1), (2,2) and (2,3). The third and final pass generates values of (3,1), (3,2) and (3,3). You now have set nine locations with subscript values from (1,1) to (3,3).

Saving Bytes

This might be a good time to conserve some bytes. Having learned to program in

✓Reader Service—see page 274

"You were rather limited in search abilities for Address Book. But look what you can do now..."

4K of RAM, I am very stingy with memory space. You can streamline this program using a couple of the features of the TRS-80. First you can write multiple lines separated by colons to combine lines 10 and 20, lines 30 and 40 and lines 50 and 60. Then, Level II allows you to use NEXT without a variable name. As long as the loops are nested properly the computer will know where to go. The compressed version follows:

```
10 FOR S=1 TO 3:FOR M=1 TO 3
20 PRINT"ENTER FIGURE FOR STORE";S;"MONTH";M:
   INPUT A(S,M)
30 NEXT:NEXT
```

When programs get long, a byte saved is a program earned (finished).

Enter the three lines and RUN them. The computer asks for exactly what it wants, then tucks the data into the right place and asks for more. Finally it is doing things my calculator can't.

To get a printout add the following lines:

```
40 Same as line 10
50 PRINT A(S,M)
60 Same as line 30
```

Try ending line 50 with a semicolon or a comma. Fool around with the PRINT@, PRINT TAB and/or PRINT USING instructions in conjunction with the printout routine. It's a snap to get neat, professional results.

You were rather limited in search abilities for Address Book. But look what you can do now:

1. To find return amounts for store 3 in the second month, enter PRINT A(3,2)
2. To find total returns for store 2 for all three months, enter PRINT A(2,1) + A(2,2) + A(2,3)
3. To find total returns for all three stores in February, enter PRINT A(2,1) + A(2,2) + A(3,2)
4. To find the difference in returns of store 1 in March and January, enter PRINT A(1,3) – A(1,1)

Getting a hint of the possibilities? Simple routines can be written to do all of these things. Here is a routine to find the total amount returned by all stores over the three month period:

```
100 FOR S=1 TO 3:FOR M=1 TO 3
110 T=T+A(S,M)
120 NEXT:NEXT
130 PRINT"TOTAL THREE MONTH RETURNS FOR ALL
    STORES";T
```

How about a routine to find the store with the most returns for the whole period?

```
100 FOR S=1 TO 3:FOR M=1 TO 3
110 IF A(S,M)>G THEN G=A(S,M):S1=S:M1=M
120 NEXT:NEXT
130 PRINT"STORE #";S1;"HAD THE GREATEST RE-
    TURNS IN MONTH NUMBER";M1
```

At the end of line 110, S1 is set to the value of S and M1 is set to the value of M. This is a reminder which store and month had the greatest amount for use at the end of the routine.

These lines merely scratch the surface of what multi-dimensional arrays can do. With larger groups of data, the flexibility and convenience is easy to imagine. I repeat that, due to the number of elements in the examples, the DIM statement was not needed. For an array with 20 rows and 18 columns, the statement DIM A(20,18) would have to be inserted before using the array. This simple statement would provide 360 locations.

Visualizing arrays with more than two dimensions is sometimes difficult. Consider the following: A financial report is presented in four volumes (one for each zone). Each volume has one page per district and the greatest number of pages in any volume is seven. Each page has a row for each store and the greatest number of rows on any page is 12. There are three columns of data for each store. To find a particular figure you must use four directions: volume number; page number; row, and column. Storing this data would require a four-dimension array such as T(V,P,R,C), in which the individual values would be set at T(4,7,12,3). The loading loop would look like this:

```
10 DIM T(4,7,12,3)
20 FOR V=1 TO 4:FOR P=1 TO 7:FOR R=1 TO 12:
   FOR C=1 TO 3
30 INPUT T(V,P,R,C)
40 NEXT:NEXT:NEXT:NEXT
```

Nested loops always end in reverse order. If they are out of order, the computer will locate to the wrong place and the program will crash. With Level II it's best to drop the loop names and let the computer figure it. That's what it gets paid for. Before you try the above routine, however, I must

"You should have known that you couldn't get out of here without homework."

warn you that it provides 4•7•12•3 or 1008 locations. Putting in data might consume some time. If you want to try it out quickly, change line 30 to read:

```
30 X=X+1:T(V,P,R,C)=X
```

This will store consecutive numbers from one to 1008 in sequence in the various locations. Use the following to get a printout:

```
50 FOR V=1 TO 4:FOR P=1 TO 7:FOR R=1 TO 12:FOR
   C=1 TO 3
60 PRINT"THE VALUE IN VOLUME";V;"PAGE";P;"ROW";
   R;"COLUMN";C;"IS";T(V,P,R,C)
70 NEXT:NEXT:NEXT:NEXT
```

It will take about 25 seconds for the numbers to load, and then almost two minutes to print out all the information even though it is scrolling rapidly up the screen. The search possibilities are varied. Listing 3 is only one example. Line 30 loads random numbers between one and 5,000 in all of the 1008 array locations. Line 60 searches all the locations for the largest number. After all locations are searched, line 80 prints out the result.

I could ramble on about complex arrays, but when you reach the point where everything in Listing 3 is clear, you won't need any more help. The best way to learn is to experiment. You should have known that you couldn't get out of here without homework. Here is your assignment:

1. Find out if the largest number is generated more than once, and if so, how many times.
2. Print the location(s) in which the largest number appears.
3. Find out if the number 238 appears in the stored numbers, and if so, where it is stored.

```
10 CLS:DIM T(4,7,12,3):PRINT @ 456,"HANG ON - THIS WILL
     TAKE ABOUT 50 SECONDS!"
20 FOR V=1 TO 4:FOR P=1 TO 7:FOR R=1 TO 12:FOR C=1 TO 3
30 T(V,P,R,C) = RND(5000)
40 NEXT:NEXT:NEXT:NEXT
50 FOR V=1 TO 4:FOR P=1 TO 7:FOR R=1 TO 12:FOR C=1 TO 3
60 IF T(V,P,R,C) > G THEN G=T(V,P,R,C)
70 NEXT:NEXT:NEXT:NEXT
80 CLS:PRINT"THE LARGEST NUMBER FOUND IS";G
```

Listing 3

Taboos

There are some taboos connected with array usage. The most frequent is forgetting to dimension the array with the DIM statement when needed. This will give you the old ?BS error message.

Once an array has been set, you may not re-dimension it. If you try, you will get a ?DD error message (One of my students maintains this stands for "dumb dimensioning."). Set arrays correctly the first time and put them near the beginning of your pro-

"The only real way to learn about arrays and matrices is to use them. So onward and upward. . ."

gram. There are methods of transposing figures from one array to another (see the third subroutine on page 6/5 of your manual), but this is a tricky process.

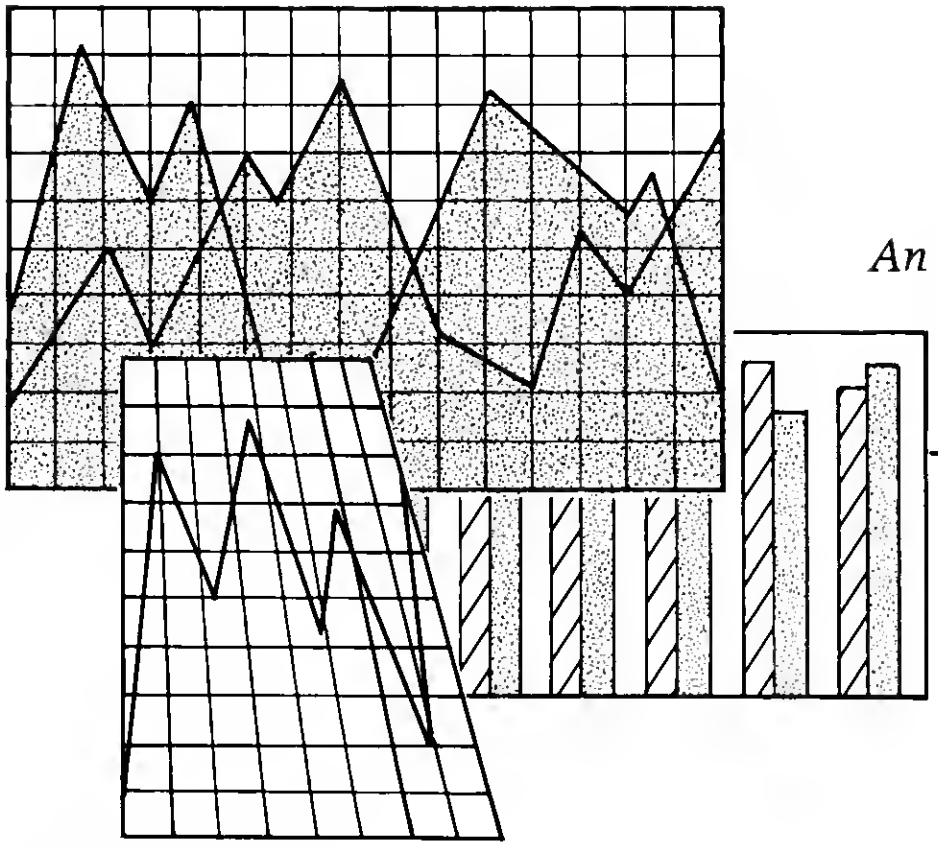
Before closing, some mention should be made of matrix operations. In algebra, tables of date are called matrices. A special branch of algebra deels with manipulating matrix information. If you are in e positlon to need scalar multiplication, element-wise functions and the like, you surely know enough to use the subroutines on pages 6/4 to 6/6 of your manual.

To demonstrate a simpler use of matrices, that of matrix addition, you should first construct another table like the Merchandise Returned table for the previous year and with different figures. The program given in Listing 4 will load date for the first year in A(S,M), load data for the second year in B(S,M) and then add the individual elements in both matrices. Lines 10 to 40 load the first matrix, lines 50 to 80 load the second and lines 90 to 110 do the addition. This creates a new matrix, C(S,M) to store the sums. Instructions are included in the program to format the printout. The new location C(1,1) contains the sum of locations A(1,1) and B(1,1). All other locations follow the same pattern.

This short tour of arrays is certainly not meant to pass as a complete treatment. It is meant to help you get from one place to another, and if it does, well and good. The only real way to learn about arrays and matrices is to use them. So onward and upward—make yourself some outrageous arrays. ■

```
10 CLS:PRINT"FIRST YEAR"
20 FOR S=1 TO 3:FOR M=1 TO 3
30 PRINT"ENTER FIGURE FOR STORE";S;"MONTH";M:INPUT A(S,
   M)
40 NEXT:NEXT
50 CLS:PRINT"SECOND YEAR":PRINT
60 FOR S=1 TO 3:FOR M=1 TO 3
70 PRINT"ENTER FIGURE FOR STORE";S;"MONTH";M:INPUT B(S,
   M)
80 NEXT:NEXT
90 FOR S=1 TO 3:FOR M=1 TO 3
100 C(S,M) = A(S,M) + B(S,M)
110 NEXT:NEXT
120 F$="$$#,###.##"
130 CLS:PRINT"THE COMBINED FIGURES ARE SHOWN BELOW:"
140 PRINT:PRINTTAB(30)"JANUARY";TAB(40)"FEBRUARY";TAB(5
    0)"MARCH":PRINT
150 FOR S=1 TO 3
160 J=27
170 PRINT"STORE #";S;
180 FOR M=1 TO 3
190 PRINTTAB(J)USINGF$;C(S,M);:J=J+10
200 NEXT:PRINT" ":NEXT
```

Listing 4



An upfront review of two statistical analysis software packages.

STATS

Advanced Statistical Analysis
Radio Shack
Ft. Worth, TX
16K Level II
Price: $39.95

Advanced Statistics
Creative Computing Software
Morristown, NJ
16K Level II
Price: $24.95

Robert P. Johnson
7547 24th Avenue, N.E.
Seattle, WA 98115

Because I believe a review should contain some recommendation, I'll tell you up front that the Radio Shack statistical package is good and worth the price. The Creative Computing package promises much, but fails to deliver.

The Radio Shack package comes in an 8½ × 11 three-ring binder, which contains a 170-page user manual and eight cassettes, including one blank. The Creative Computing package, in an 8½ × 5½ vinyl folder, contains a 27-page user booklet and one cassette.

The Radio Shack software supports printed output. The manual includes a sample of the video display at each program stage and a discussion of possible error messages and other user foul-ups. The manual also has four appendices, one containing program listings.

The Creative Computing booklet only describes how to use the statistical programs.

CLOAD Cassettes

A short digression: I recommend that you recopy the Creative Computing programs onto another cassette. On the original cassette, the programs are recorded one after the other, so if you wish to use the seventh program, you must CLOAD seven times. Copy the programs onto another cassette and give each a unique file name. Or load each onto one side of a single cassette.

Writing this review, I discovered that the programs on the Advanced Statistics cassette *do* have file names. This is not mentioned in Creative's booklet. The file names are not the numbers one to nine as you might expect, but the letters A to I. So, to load the Multiple Linear Regression program (number 6 on the table of contents), you would type CLOAD F. This works only if you start from the beginning of the tape.

The Programs

Both packages contain roughly equivalent statistical programs. The Radio Shack package contains two programs that are absent from the Creative Computing package, and the Creative package contains one program that is not in Radio Shack's. The Radio Shack Random Sample program selects a random sample of data item numbers from a population. The second program is Time Series II. It calculates seasonal indices and moving averages for yearly, quarterly, monthly, weekly or daily data. In my work I find little use for the random sample program, but the Time Series II is invaluable.

The extra program in the Creative Computing package is Correlation Analysis. It performs correlation analysis on up to five variables. The statistical output of the program is similar to that of the multiple regression program, but may prove to be useful to users who need correlation analysis.

While the remaining programs in both packages are similar, differences exist and will be covered in the following discussion.

Most of the programs do not contain data correction routines. To minimize your frustration, it is essential to load your KBFIX program first. This won't solve all of the user input errors, but it will reduce them to the minimum acceptable aggravation level.

Tape Data Files (RS) and Data File Manager (CC)

These programs are the heart of a data management system for a statistical analysis package. They allow data to be stored on tape for repetitive use and allow that data to be edited. The Radio Shack program can be used to create new data files, list data files (with a printer option), or update old data files by deleting or adding data elements for existing variables, or by adding a new variable and its data elements.

The Creative Computing package does this and more. This program allows the user to create a new file containing only some of the variables on the master file; to substitute (delete and add combined) values for the variables on the master file; to perform transformations of variables on the master

"To minimize your frustration,. . . load your KBFIX program first. This won't solve all. . . errors, but it will reduce them to. . . acceptable aggravation level."

file; and to create a subfile containing some of the values of the variables on the master file. A further word about the transformation option: the program allows any variable to be transformed as follows:

New Variable = INT((Old Variable + A)*D)

Here the values for A and D are user-supplied. The user's booklet tells you which program line to change to allow other transformations. Nice touch.

Both packages require that different statistical programs use different data file formats. You cannot create a data file and run it with eny of the programs. Statistical packages for mainframe computers (of which SPSS—Statistical Package for the Social Sciences—is probably the best known) generally allow the user to use a common data file format for all of the statistical analysis routines. This is helpful and I hope that the next generation of personal computer statistical software writers will adopt the method.

Descriptive Statistics, Histogram, and Frequency Distribution from Radio Shack, as well as Descriptive Statistics from Creative, are programs for the statistical analysis of a single variable. It should be mentioned that Histogram (RS) is really a graphics program and not useful unless printed.

The two packages produce comparable information, except that the Creative Computing program hes more features than the Radio Shack programs. It has two options that are used for test scoring. You enter either the number of questions right or the number of questions wrong, and the progrem scores each test, producing a statistical analysis of the scores.

Both packages produce similar descriptive statistics. The Creative Computing program includes the median, quartile values, and the standard error of the mean. The latter can be calculated using output of the Radio Shack program, but the first two values cannot.

Descriptive Statistics from Creative can correct erroneous data entries before running the program, but I had problems with this option. When I deleted data, even though it did not show up in the revised data listing, it was still part of the statistical computations.

T-test for Matched Pairs, and Correlation and Linear Regression from Redio Shack perform much the same job as Two Variable Statistics from Creative. Both packages perform standard two-variable statistical analysis. There are two major differences between them, however. The Creative Computing program conducts only a two-tailed t-test on the data, while the Radio Shack programs allow either one- or two-tailed t-test. Also, the Radio Shack programs graphically display the two-variable regression and the original data points on the screen (or the printer).

The Radio Shack program, however, does not calculate the standard error of the estimate for the regression. The Creative Computing program does. Since both packages allow forecasting of the dependent variable, Radio Shack's omission is puzzling.

Chi-Square Analysis from Radio Shack and Crosstabulation from Creative are basically similar. The first major difference is the dimensions of the chi-square table. The Radio Shack program accepts up to an 8 × 8 matrix; the Creative Computing program accepts up to a 10 × 10 matrix. Other differences:

- The Redio Shack program allows the user to specify the expected cell frequences.
- The Creative Computing program allows the chi-square matrix to be consolidated into a 2 × 2 contingency table, a useful feature if some frequencies are low or missing.
- The Creative Computing program allows the data to be entered raw. Each observation is entered as row and column numbers and the program then calculates the observed frequencies. You would be better off (in terms of finger fatigue) to calculete the frequencies before using the program.
- The Creative Computing program computes a gamma statistic in addition to the chi-square.

Regression-trend Analysis from Creative is easy to use as the time variable is abstract (period 1, 2, 3, etc.) and is automatically incremented with each data entry for the dependent variable. It also estimates the regression coefficients for eight functional forms, (including the linear model). Unfortunately, no information is provided so thet the user can determine which functional form is best fit to the data (aside from the standard error of the estimate). Neither the regression routine in Creative's Two-Variable Statistics nor their Regression-trend Analysis calculates a correlation coefficient.

Multiple Linear Regression and Advanced Multiple Regression in the Creative Computing peckege seem to be e "band-aid." Multiple Linear Regression is compatible with the Data File Manager while the Advanced Multiple Regression program is not. The latter only accepts data from tape in the form of DATA statements appended to it. The new program can be recorded on tape for later use. This is not a flexible system. The Multiple Linear Regression program, like the rest of the programs in the package, was written by Richard Galbraith, while Advanced Multiple Regression was written by David J. Simecek.

Why a different author?

So what is wrong with the multiple regression program written by Galbraith?

First: The output consists of partial correlation coefficients between variable pairs, the means and standard deviations of the variables, and two sets of the regression coefficients. One set is the regular equation with no intercept, or constant, term. And that is it! Now I don't know of anyone who would estimate a regression equation and force the constant term to be zero.

The second problem with the Galbraith program is that it is unreliable. In testing all three programs, I used Multiple Regression Analysis—Simplified, an article by Dr. David M. Chereb in the February, 1979, issue of *Creative Computing*, as a benchmark. When I used the data with Creative's Mulitple Linear Regression program, it ran through the correlation matrix and then produced the message:

THERE IS NO UNIQUE SOLUTION

I suspect that the matrix inversion algorithm produced a singular matrix. This is flatly unacceptable since the data does produce a solution in other programs.

The Advanced Mulitple Regression output consists ot the regression coefficients, calculated t-values, a calculated F statistic, confidence intervals for the regression coefficients based on user-supplied t-values, and analysis of variance table, and the coefficient of determination (R^2)—mislabeled as the "coefficient of multiple determination."

The program contains a data review and correction option, but after displaying the values of the dependent variable, the screen prompt reads: CORRECT AS FOR Y VALUES?. A response of YES gets a REDO message. The correct response is 0,0 as with the independent variable correction routine.

Another irritant is the prompt TYPE 1 FOR ANOTHER SET?. What do you type if you *don't* want another set of estimates? Answer: any other number; but that's not obvious. With some experimentation you can clear the problems up. That such "minor" problems exist, however, is evidence that the program has not received extensive user testing.

One major problem surfaced when I used the data from Dr. Chereb's article. The pro-

gram produced the correct regression coefficients, but then went on to generate an erroneous analysis of variance table and other summary statistics. For example, the calculated F statistic was reported as −32.41 (it is actually 1594.62), the R^2 was reported as 1.33 (it is actually 0.995), and the t-values were also incorrect.

While Radio Shack's Multiple Linear Regression program performs well (it never missed a beat with any data I fed it—including Dr. Chereb's infamous set), and while it produces an acceptable set of summary statistics, it doesn't have some of the features that one might like. For example, it only handles up to five independent variables.

Also lacking in the program's summary statistics are t values or standard errors for the regression coefficients, a standard error of the equation, and an R^2 adjusted for degrees of freedom. None of these are difficult to include, and they are indispensable for hypothesis testing.

While the Radio Shack and Creative Computing packages contain programs for Analysis of Variance, Radio Shack's program is considerably more limited, to a one-way analysis from two to five groups or samples. Radio Shack also includes an addendum page, explaining which two lines of the program are to be changed. The error is non-fatal as it occurs in the program segment that controls the printing of the test statistics to the screen or the printer. The addendum is a nice touch, however, and shows some concern for testing of the programs.

Two-Way Analysis

The Creative Computing program will run one and two-way analyses of variance, up to 10 or 11 groups, one-way, (up to 100 or 121 groups' two-ways).

Creative's one-way ANOVA performs faultlessly and its output is perhaps better because the program generates sample means and standard deviations for each group and the sample as a whole. The two-way ANOVA is another story. After it generates the mean and standard deviations, the error message: /0 ERROR IN 2830 appears. Line 2830 reads:

```
2830  Y = 2/(9*B) : X = 2/(9*A)
```

Somehow, somewhere, A or B are either being set to zero or have not been set after initialization, causing the program to bomb. As with both of the multiple regression programs, you could (if you have access to a printer) get a printout of the program and attempt to debug it.

This, of course, is the point. The programs are sold (and advertised) as complete, ready-to-run software. They are not. ■

MODEL II

26-4002
64K 1 Drive
$3466.00

MODEL III

26-1061 4K I.....$630.00
26-1062 16K III.....900.00
26-1063 32K III
2-Drives, RS232.....2246.00

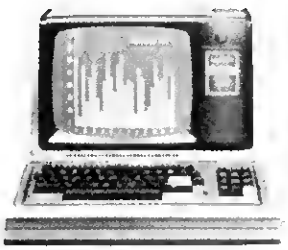
DISCOUNT TRS-80® DEALER A301

COMPUTER SPECIALISTS

26-1051 4K Level I System.....$424.00
26-1054 4K Level II System.....552.00
26-1145 RS-232 Board.....84.00
26-1140 "O" K Interface.....249.00
26-1141 "16" K Interface.....359.00
26-1142 "32" K Interface.....469.00
26-1160 Mini Disk - Drive O.....419.00
26-1161 Mini Disk - Additional.....419.00
26-1154 Lineprinter II.....699.00
26-1156 Lineprinter III.....1799.00
26-1159 Lineprinter IV.....859.00
26-1104 Factory Upper/Lower Case Modification Installed.....70.00
26-1506 Scripsit - Tape.....60.00
26-1563 Scripsit - Disk.....79.00
26-1566 Visicalc.....83.00
26-1562 Profile.....72.00

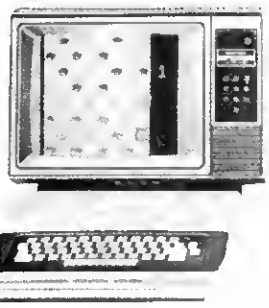
NOTE: Call for availability of VIDEO TEX, Model III, Color, and other new products.

MODEL I

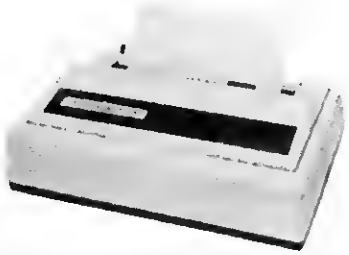


26-1056
16K Level II System
$670.00

COLOR



26-3001 4K.....$360.00
26-3002 16K.....540.00
26-3010 Color Video.....360.00
26-1206 Recorder.....54.00
26-3008 Joysticks.....22.50



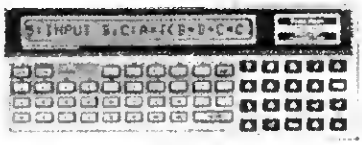
CENTRONICS

Fast 100 CPS Centronics
730 Printer.....$659.00
Text Quality Centronics
737 Printer.....$819.00

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES DEDUCT 10% FROM CATALOG PRICE

Novation Cat Modem..$149.00
CCA Data Management System.....72.00
Adventure Games Games 1-9 each.....14.00

Packet Computer



26-3501 1.9K P.C.....$225.00
26-3503 Cassette I/F.....45.00
14-812 Recorder.....72.00

Acorn Software Products, Inc.

GAMES:
Alien Invasion.....$9.00
Stock Market.....9.00
Star Trek.....9.00
Block 'Em.....9.00
Ting-Tong.....9.00
UTILITIES:
System Savers.....14.00
EDUCATION:
Language Teacher.....18.00

FREE: COMPUTER CATALOG UPON REQUEST

Model II Cobol Compiler
$360.00
Cobol Run Time Package
$36.00

1-800-841-0860 Toll Free Order Entry

✓72

MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of State Shipments

Immediate Shipment From Stock on Most Items

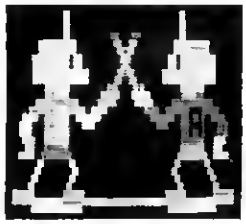
DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

R.S. 90 Day Limited Warranty
F-48 Form Provided

Largest Inventory In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp.

THE PROGRAM STORE



new

DUEL «N» DROIDS

By Leo Christopherson from Acorn
Your 'droid has already learned NIM, so now it's time to teach it how to wield a laser sword! Leo Christopherson, author of "Android NIM," "Dancing Demon" and other animations, has developed a new type of animation and high-quality sound in his latest work.

Your 'droid starts out as a lowly clown. You teach it how to use a laser sword by controlling its movements. After training it to be a "Grand Master," you enter the tournament against the program's skilled 'droid! Entertainment for all ages.

Protected Tape...$14.95
Protected Disk...$20.95

DEATH-MAZE 5000

from Med Systems
A new breed of adventuring! Venture through a graphically represented 3-D maze, with halls that could dead end -- or recede to infinity. Step through the doors or drop into the pits. Will you encounter monsters and mayhem, or will you be treated to useful objects and information? Will you ever get out alive?

You may never find your way out of Deathmaze 5000, but you'll keep trying!

16K TRS-80, 32K APPLE II...$12.95

HAIL to the CHIEF

By P. Brasher & R. Vance from Sensetional Software
How would you run a political campaign for the highest office in the land? Would you be elected? Find out with this campaign strategy simulation developed by political scientists. Choose (and perhaps change) your positions on major issues as you conduct your campaign, all the while keeping an eye on the weekly polls.

TRS-80 32K Cassette, 48K Disk
Apple II & Apple II+ 48K Disk
Atari 400 32K Cassette, Atari 800 40K Disk
.....$24.95

TRS-80 Level II 16K unless otherwise noted

new BASKETBALL Dribble Dribble

By John Allen from Acorn
New machine language action game, with sound, from the author of the acclaimed "PINBALL"!

You have to be fast to keep up with the action as you try to outscore your opponent in five minutes of one-on-one basketball. Compete against a friend or your computer.

Steal the ball, duck around your opponent and slant toward the basket for a lay up! The graphics are based on a 3-dimensional depiction of a basketball court, and ball dribbling sounds add to the realism. It's all there but the cheers -- so real you'll wonder how the ball keeps from coming through the screen of your TRS-80! Dribble, Dribble!

Protected Tape...$14.95
Protected Disk...$20.95

GALACTIC TRILOGY

By Douglas Carlston
Take control of the Galactica as you navigate through an uncharted 3-dimensional universe. In "Galactic Empire," you attempt to unify a universe that is randomly created each time you play.

"Galactic Trader" pits your bartering skills against those of the other inhabitants as you try to accumulate riches and power. But watch out for the assassins and the energy cartel -- they're out to getche!

Diplomacy and deviousness play equal parts in "Galactic Revolution." It's a game that combines tactics, social manipulation and Machiavellian ruthlessness. For more intrigue, this game allows more than one player. Sound effects.

Choose any game at $14.95 for TRS-80 16K on tape. $24.95 for Apple II 48K Disk.

To control the entire universe, get all three!

JET FIGHTER PILOT

from Instant Software
Launch one of several realistic jet fighters from an airport, or catapult from the deck of an aircraft carrier. Incredibly realistic simulation, right down to maintenance problems.

You will not only learn about the dynamics of flight, you'll discover the complex operation of modern military jet aircraft as you sit back and try to keep up with the constantly chenging instrument panel display. Challenging and informative.

Cassette...$14.95

EDAS Editor/Assembler

By Roy Soltoff from MISOSYS
With EDAS, you are no longer tied to memory limitations while writing in assembly language. Now you can assemble directly from text stored on disk. Branching lets you test your program, then return directly to EDAS. Great for editing and debugging.

Other features include: global editing, upper/lower case support, block moves, plus availability of DOS commands within EDAS. It's the Editor/Assembler designed with the programmer in mind!

Disk...$79.00

MiGs & Messerschmidts

"It is the summer of 1941 and the Blitzkreig is smashing into the heart of Russia..."

This is how your instructions begin when you become the fighter squadron leader in "Migs & Messerschmitts", one of four exciting new Discovery Air Combat Simulations.

These World War II re-enactments are historically accurate -- they challenge you to learn the tactics used by the actual combatants! Written in machine language for fast response.

MIGs and Messerschmidts
RAF: The Battle of Britain
Jagdstaffel
Winged Samurai

For TRS-80, Apple II, PET -- 16K...$19.95

ACCEL & ACCEL II

From Allen Gelder Software
Imported from England, a compiler for TRS-80 Level II Basic (ACCEL) and Disk Basic (ACCEL II). ACCEL lets you compile the integer portion of your Basic programs to fast, efficient Z-80 machine code. ACCEL II compiles floating-point arithmetic as well, and supports Disk Basic.

Both allow a significant improvement in run-time -- up to 3000% faster in some cases -- and improved program security!

ACCEL.....$49.95
ACCEL II....$89.95

Visit Our New Store: W. Ball Plaza – 6600 Security Blvd · Baltimore, MD

VISA MasterCard **TO ORDER CALL TOLL FREE 800 424-2738** For information Call (202) 337-4691

THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept. K6 Box 9609
Washington, D.C. 20016

MAIL ORDERS: Send check or M.O. for total purchase price, plus $1.00 postage & handling. D.C. residents, add 5% tax. Charge card customers: include all embossed information on card.

✓ Reader Service—see page 274

PROGRAMMING TOOLS FOR YOUR TRS-80

INSIDE LEVEL II

The Programmers Guide to the TRS-80 ROMS

INSIDE LEVEL II is a comprehensive reference guide to the Level II ROMs which allows the machine language or Basic programmer to easily utilize the sophisticated routines they contain. Concisely explains set-ups, calling sequences, and variable passage for number conversion, arithmetic operations, and mathematical functions, as well as keyboard, tape, and video routines. Part II presents an entirely new composite program structure which loads under the SYSTEM command and executes in both Basic and machine code with the speed and efficiency of a compiler. In addition, the 18 chapters include a large body of other information useful to the programmer including tape formats, RAM useage, relocation of Basic programs, USR call expansion, creating SYSTEM tapes of your own programs, interfacing of Basic variables directly with machine code, a method of greatly increasing the speed at which data elements are stored on tape, and special precautions for disk systems. INSIDE LEVEL II is a clearly organized reference manual. It is fully typeset and packed with nothing but useful information. It does not contain questions and answers, ROM dumps, or cartoons. **INSIDE LEVEL II..... 15.95**

TELECOMMUNICATIONS PROGRAM

This program allows reliable high speed file transfers between two disk-based computers over modems or direct wire. It is menu driven and extremely simple to use. Functions include real-time terminal mode, save RAM buffer on disk, transmit disk file, receive binary files, examine and modify UART parameters, program 8 custom log-on messages, automatic 16-bit checksum verification of accurate transmission and reception, and many more user conveniences. Supports line printers and lowercase characters. With this program you will no longer need to convert machine language programs to ASCII for transmission, and you will know immediately if the transmission was accurate. **TELCOM.....$29.95**

PROGRAM INDEX FOR DISK BASIC

Assemble an alphabetized index of your entire program library from disk directories. Program names and free space are read automatically (need not be typed in) and may be alphabetized with a fast Shell/Metzner sort by disk or program. The list may also be searched for any disk, program, or extension; disks or programs added or deleted; and the whole list or any part sent to the printer. Finally, the list itself may be stored on disk for future access and update. "The best thing since sliced bread" (January issue of '80 Microcomputing). One drive and 32K required. **INDEX.....$19.95**

SINGLE STEP THROUGH RAM OR ROM

STEP80 allows you to step through any Basic or machine language program one instruction at a time, and see the eddress, hexadecimal value, Zilog mnemonic, register contents, and step count for each instruction. The top 14 lines of the video screen are left unaltered so that the "target program" may perform its display functions unobstructed. STEP80 will follow program flow right into the ROMs, and is an invaluable aid in learning how the ROM routines function. Commands include step (trace), disassemble, run in step mode at variable step rate, display or alter memory or CPU registers, jump to memory location, execute a CALL, set breakpoints in RAM or ROM, and relocate to any page in RAM. The display may also be routed to your line printer through the device control block so custom print drivers are automatically supported. **STEP80.....$16.95**

4 SPEED OPTIONS FOR YOUR TRS-80!

The SK-2 is the most versatile clock modification available for the TRS-80. Speeds may be switched between normal, an increase of 50%, or a 50% reduction; selectable at any time without interrupting execution or crashing the program. Instructions are also given for a 100% increase to 3.54 MHz, though the TRS-80 is not reliable at this speed. The SK-2 may be configured by the user to change speed with a toggle switch or on software command. It will automatically return to normal speed any time e disk is active, requires no change to the operating system, and has provisions for adding an LED to indicate when the computer is not at normal speed. It mounts inside the keyboard unit with only 4 necessary connections for the switch option (switch not included), and is easily removed if the computer ever needs service. The SK-2 comes fully assembled with socketed IC's and illustrated instructions. **SK-2.....$24.95**

RAM SPOOLER AND PRINT FORMATTER

This program is a full feature print formatting package featuring user defineable line and page length (with line feeds inserted between words or after punctuation), screen dump, and printer pause control. The serial version allows baud rate selection from the keyboard. In addition, printing is done from a 4K expandable buffer area so that the LPRINT or LLIST command returns control to the user while printing is being done. Ideal for Selectric or other slow printers. Allows printing and processing to run concurrently. Please specify PARALLEL or SERIAL (RS-232 interface) version. **SPOOLER.....$16.95**

DUPLICATE SYSTEM TAPES WITH CLONE

Make duplicate copies of ANY tape written for Level II. They may be SYSTEM tapes (continuous or not) or data lists. The file name, load address, entry point, and every byte (in ASCII format) are displayed on the video screen. **CLONE.....$16.95**

MACHINE COOE FAST FOURIER TRANSFORM

This complete package includes 3 versions of the machine language FFTASM routine assembled for 16, 32, and 48K machines, a short sample Basic program to access them, a 10K Basic program which includes sophisticated interactive graphing and data manipulation, and a manual of instructions and examples. The machine language subroutines use variables defined by a supporting Basic program to make data entry and retrieval extremely fast and easy for custom implementation. They perform 20 to 40 times faster than their Basic equivalent (256 points in 12.5 seconds), and require less than 1550 bytes of memory. **FFTASM.....49.95**

FOR THE MODEL II

LYNC

from Midnight Software

High level data communication for the Model II with CP/M. LYNC will send and receive any file with automatic error checking and retries. Either end may initiate file transfers, and multiple files may be sent with wildcard filenames. Remote or local directories may be called from within the program. Allows full protocol, non-protocol, and real-time conversation modes. May be used over phone lines at 300 baud or direct to another computer at up to 9600 baud. Also available for other CP/M computers. **LYNC.....$95.00**

MUMFORD MICRO SYSTEMS

ORDERING: Complete satisfaction is guaranteed or refund will be made. All Models programs are shipped on diskette unless $5 is included for formatted (no system) disk. Include $1 postage and handling. California residents add 6% sales tax. Visa, MasterCharge and C.O.D. orders accepted.

Box 435-E Summerland, California 93067 (805) 969-4557

A successful business application from beautiful, downtown Burbank.

The Office Computer

Gery Valle
7219 Loma Verde
Canoga Park, CA 91303

Yes sir, I can check that order for you. Would you please give me your purchase order number?

The secretary turns to the computer console and enters the number. A moment passes, and the CRT displays the order.

"Sir, that was shipped 3/05/80 . . . Yes, 103 of Pretlight Procedures, 105 of Airworthiness Testing and 50 of Stability—Part I."

Is this the efficient customer service department of a *Fortune* 500 business? Not quite, or, at least, not yet.

Film Systems, Inc. is a growing company that specializes in the production and duplication of slides and filmstrips for educational institutions, business and industry. FSI uses a microcomputer as a tool to control its own success and its expanding clientele.

Information Processing

Large or small, all businesses have at least one characteristic in common—they process information. Whether it's an inquiry, an order, market research, or the weekly payroll, any business devotes a considerable portion of its time and money to processing words and numbers.

If sales are to be increased, more orders are to be received, purchases made, journal entries posted, invoices mailed and correspondence written additional information processing is a necessity. Why does the burgeoning small business so often overlook this requirement? The business may employ many factory workers, a production manager, a designer, two or three sales people, and be considering hiring others, and yet its office is already understaffed and overworked.

Often the office is still a one person show. That one individual opens the mail, processes orders, types correspondence and invoices, answers the phone, receives customers, figures the payroll and taxes, does the bookkeeping, sends out statements, checks on overdue supplies, makes up the bank deposits, keeps the company checkbook up-to-date and more. As a business grows, these responsibilities can easily overpower even the most productive individual. If the company's ability to process information is not improved, inquiries may go unanswered, bookkeeping may fall behind, orders may not be shipped when promised and inventory may not be properly maintained.

Early on FSI management recognized that a well-organized and productive office is essential to its success and continued growth. On the left of the secretary's desk at FSI is a typewriter—on the right is a 48K TRS-80 with a tractor feed line printer and dual disk drives.

FILM SYSTEMS INC.
1633 North Mena Street • Burbank California 91504
Telephone 213 / 841-3370

INVOICE
3348

SUNBIRD GLIDERS
12501 GLADSTONE #A4
SYLMAR CA 91342

SUNBIRD COSTA MESA
1760 MONROVIA #86
COSTA MESA CA 92627

TERMS: NET 30

3/05/80	2295	9861	
100 FRS	MAS NEG FROM SLIDES-STABILITY PART I	4.00	400.00
1 EA	CRI/FIRST-THERMAL SOAR	40.00	40.00
3 FRS	PICKUP/SLIDE MASTER NEG-THERMAL SOAR	4.00	12.00
	TAXABLE TOTAL		452.00
	LESS 5 PERCENT DISCOUNT		22.60
	DISCOUNTED TAXABLE TOTAL		429.40
	6% CALIF SALES TAX		25.76
3 HRS	TECH LABOR-STABILITY PART I	25.00	75.00
1000 FT	RELEASE PRINT-RIDGE SOARING	0.175	175.00
100 PRT	CUT/CAN-RIDGE SOARING	0.08	8.00
	NON-TAXABLE TOTAL		258.00
	LESS 5 PERCENT DISCOUNT		12.90
	DISCOUNTED NON-TAXABLE TOTAL		245.10
	SHIPPING		2.46
	INVOICE TOTAL		$702.72

Fig. 1. Invoice Generated on TRS-80.

Photo 1. Order Displayed on CRT for Servicing Customer Inquiries.

Photo 2. Bar Graph Displaying Sales by month.

Unfortunately the word "computer" still brings to mind an omnipotent machine of unmanageable size and tempermental nature, consuming punched cards and spewing forth great volumes of paper or magnetic tape. It would be more accurate to think of a microcomputer as a wrench or hand drill! The computer is a tool. It is simply a machine that can process, store and retrieve information quickly and repetitively.

Invoices

The first FSI bottleneck was invoice writing. The office was spending hours and hours writing invoices. Three factors contribute to this manual inefficiency:

- Much of the information used in processing invoices day-to-day is repetitive.
- Lists must be searched for service or parts names and prices.
- Extensions, totals, discounts and taxes must be calculated.

A microcomputer is well suited to these tasks. Customer names, billing addresses, shipping addresses, service and product titles, part numbers and prices and other information can be stored on disk files. This information can be accessed and typed very quickly. For example, consider the operator input necessary to write the invoice shown in Fig. 1.

Responding to prompts, the operator enters the data, customer number, ship-to option, customer reference number and job number:

```
DATE?: 3/05/80          (entered only at beginning of session)
CUSTOMER NUMBER?: 44
SHIP-TO OPTION?: 2
FSI JOB NUMBER?: 2295
CUSTOMER REFERENCE?: 9861
```

After printing the invoice headings the next prompt is:

```
TITLE?: STABILITY-PART I          (optional film title)
SERVICE CODE?: 1
QUANTITY?: 100
```

Referring to a menu of services on the video display the operator enters a code and the quantity to be invoiced. The invoice writer does in minutes what used to take hours.

Another log jam in the FSI information flow was order processing. A large percentage of FSI orders are for filmstrips. The filmstrip is made from a previously processed master negative. These negatives are cataloged by title and are assigned a production

Photo 3. Display Monitoring Level of Activity of Key Invoice Items.

Photo 4.

number as well as other data used in making the filmstrip.

In pre-computer days when an order was received, the titles specified would be looked up in the negative catalog for the appropriate production numbers, a release order listing the quantity, production number and title. This and other pertinent information would be typed in duplicate and a "shipper" in triplicate. After the order was shipped an invoice would be typed in triplicate.

Most of the information contained in the order, shipper, and invoice is the same; why type it three times? Now the negative catalog is stored on disk and up to 3000 titles can be searched by a machine language program in well under a second. With only a minimum of input from the secretary, the release orders and shippers are typed on the line printer.

A disk file of current orders is maintained and by specifying either the customer's purchase order number or the in-house job number an order can be displayed on the CRT (Photo 1). A hard copy option will print the order if desired. After the current order file has been updated with shipping information, the invoice writer previously described uses the same information in the file to write an invoice. The secretary need only specify the order number.

More Useful Results

After the major bottlenecks have been eliminated you are likely to discover that your data files and programs bear much fruit. After the initial planting, cultivation and growth, a great deal of information may be harvested with little additional effort.

Relatively simple modifications to existing software may generate useful results. With a simple addition to the invoice writer, cumulative monthly sales totals may be added to the customer information file. In turn this information can generate a bar graph that displays sales by month for any specified customer (Photo 2).

Another straightforward addition allows you to monitor the activity of certain key invoice items for a specified customer or for all customers (Photo 3). In many businesses the function of an order/invoice writer can be extended to support accounts receivable and sales journal processing.

With the addition of a purchases journal, the system can audit inventory depletion, prompt purchases when order points are surpassed, and also process accounts payable data.

The typewriter and the adding machine have for decades been the principal information processing tools of the small business. As certain as the calculator has replaced the slide rule, these traditional business machines have become outdated. I imagine there are a few stubborn individuals who will insist on using a slide rule instead of a calculator. No doubt similar individuals will continue to plug away on adding machines and standard typewriters, but for the great majority of us the course is clear. Make way for the office microcomputer! ■

✓ Reader Service—see page 274

Velerie Venn
631 G Street
Davis, CA 95616

As the holiday season approached last yeer, I thought it would be great to have something my computer could do to show off to visitors. "Battleship," "Tank Wers," and "X-Wing Fighter Bombing Runs" didn't seem appropriate. A computer Christmas card was more like what I had in mind.

These programs will run on a 16K Level II Model I, TRS-80. An alternate version of one subroutine is included for Microsoft's Level III BASIC.

A Series

The program Holiday Graphics & Seasons Greetings, (Program Listing 1), is a series of five subroutines: Snowflakes, Seasons Greetings, Poem (a Level II BASIC adaptation and enhancement of a routine from David Lien's Level I Manual), Snow Scene, and a Signature Page (for personalizing your computer greeting card). The program is designed to repeat endlessly and one complete cycle takes an average of 26 minutes.

If you're getting the family an 80 for Christmas, you could even sit it under the tree and let it run this program.

The program is numbered in modules of 1,000. Each begins with a REM statement identifying the subroutine number.

Each routine can be run alone or with one or two others by making minor changes. The first lines of the program identify initialization statements for the subroutines. The ending lines contain the RESTORE statement to run the Season's Greeting and Snow Scene routines in an endless cycle. The GOTO statement returns the program to the Snowflake routine for another complete cycle.

If you want to run the Poem routine by itself, a FOR-NEXT time delay loop should be added to the end. This is because the time used in setting up the following Snow Scene graphics array serves as a time delay in the combined program.

The displays are best viewed from a distance of at least eight feet in a dimly lit room. The brightness and contrast of the CRT should be adjusted to give a crisp black and white effect. Then sit back with your cup of eggnog and watch it snow!

The Snowflake subroutines contrast the BASIC language graphics routines with the speed and simplicity of the vector graphics enhancements in Microsoft's Level III BASIC. This uses the line plotting statement LINE(X1,Y1)–(X2,Y2),SET.

The Level II version substitutes a line plotting subroutine for this Level III statement, and thus runs slower. Both versions are compatibly line numbered so the differences can be identified readily.

Six-Sided Designs

The Snowflakes programs (see one in Program Listing 2) draw six-sided designs on the video display screen. They use the smallest TRS-80 graphics block, or pixel. Like real snowflakes, the odds against getting two alike are astronomical!

Each flake is drawn in 5 to 12 cycles of line plotting (the number of cycles is selected at random). In each cycle, the X and Y coordinates for two points are generated at random. The line defined by the two points is then plotted, and rotated in 60 degree increments and plotted in six positions. The mirror image of this rotating line is then computed and plotted.

The designs are not always symmetrical because coordinates are rounded to the nearest integer value. You will also notice a pixel of variation in some line positions.

The results are usually attractive, especially considering the limits of the TRS-80 graphics system: You are plotting hexagonal figures with rectangular blocks.

Line 270 contains an adjustment factor for the aspect ratio of the screen (constant V). This produces a round visible plot on my screen. You may wish to adjust it slightly to get the best results on your CRT. Use something in the neighborhood of 128/48.

Other kaleidoscopic effects can be produced by changing the angle of rotation (lines 550-560, P/3) and the number of plotting positions (line 540, FOR J = 1 TO 6). Change these to P/(n/2) and FOR J = 1 TO n, where n is the number of sides you want the figure to have. Also, the aspect ratio constant V can be changed or eliminated.

Trigonometry and analytic geometry teachers and their students might have fun taking the program apart. It contains all those basic elements like polar coordinates; translation and rotation of axes; slopes and intercepts.

Anyway, it's fun to watch. The snowflakes even have a crystalline appearance, thanks to the little rectangular blocks. They seem to grow like frost patterns on a window. ■

Program Listing.

```
10 REM INITIALIZE GRAPHICS ROUTINES
20 RANDOM: REM ROUTINES 2,4,5
30 CLEAR 1000: REM ROUTINE 5
40 DIM A$(16):OIM S(21,2): REM ROUTINE 5
50 CLS:PRINT"HOLIDAY GREETINGS WITH SNOW . . . ."
60 PRINT:PRINT"GRAPHICS BY VALERIE VANN"
70 PRINT CHR$(204)+"631 G ST., DAVIS, CA."
80 PRINT CHR$(204)+"COPYRIGHT 1980"
90 PRINT"POEM SUBROUTINE ADAPTED FROM A LEVEL I BASIC P
   ROGRAM"
100 PRINT"BY DAVID LIEN."
110 FOR X=1 TO 1000:NEXT X
1000 REM ENTER SIGNATURE - ROUTINE 1
1010 PRINT@512,"IF YOU WISH TO SIGN THIS GREETING, TYPE
     YOUR NAME,":PRINT@576,"THEN PRESS ENTER.  IF NOT,
     JUST PRESS ENTER."
1020 PRINT"(MAXIMUM OF 20 CHARACTERS)":PRINT"  "+STRING
   $(20,"-")
1030 INPUT B$:IF LEN(B$)>20 THEN 1020
1040 IF B$=""THEN B$="YOUR FRIENDLY COMPUTER - ME!"
2000 REM TITLE PAGE SNOWFLAKES - ROUTINE 2
2010 CLS
2020 PRINT CHR$(23)
2030 FOR J=2 TO 442 STEP 0
2040 PRINT@J,"*"
2050 FOR F=1 TO 10:NEXT F
2060 NEXT J
2070 PRINT@452,"* * S N O W F L A K E S * *"
2080 FOR J=514 TO 958 STEP 0
2090 PRINT@J,"*"
2100 FOR F=1 TO 10:NEXT F
2110 NEXT J
2120 FOR J=1 TO 950:NEXT J
2130 V=120/40
2140 P=3.141592654
2150 FOR E=1 TO 5
2160 CLS
2170 FOR K=1 TO (RND(5)+7)
2180 X=RND(24)
2190 Y=RND(24)
2200 R=SQR(X[2+Y[2)
2210 IF R>24 THEN 2180
2220 T=RND(24)
2230 Z=RND(24)
2240 S=SQR(T[2+Z[2)
2250 IF S>24 THEN 2220
2260 GOSUB 2340
2270 Y=-1*Y
2280 Z=-1*Z
2290 GOSUB 2340
2300 NEXT K
2310 FOR I=1 TO 2000:NEXT I
```

Program continues

```
2320 NEXT E
2330 GOTO 3000
2340 W=ATN(Y/X)
2350 Q=ATN(Z/T)
2360 FOR J=1 TO 6
2370 W=W+P/3
2380 Q=Q+P/3
2390 X2=R*COS(W)
2400 Y2=R*SIN(W)
2410 T2=S*COS(Q)
2420 Z2=S*SIN(Q)
2430 X3=(X2*V+64):Y3=(Y2+24):T3=(T2*V+64):Z3=(Z2+24):GO
     SUa 2460
2440 NEXT J
2450 RETURN
2460 IF X3=T3 THEN 2620
2470 IF X3>T3 THEN M=(Y3-Z3)/(X3-T3) ELSE M=(Z3-Y3)/(T3
     -X3)
2480 B=Y3-M*X3
2490 IF ABS(Z3-Y3) >ABS(T3-X3) THEN 2560
2500 IF X3>T3 THEN D=-1 ELSE D=1
2510 FOR H=X3 TO T3 STEP D
2520 Y3=M*H+a
2530 SET(H,Y3)
2540 NEXT H
2550 RETURN
2560 IF Y3>Z3 THEN D=-1 ELSE D=1
2570 FOR H=Y3 TO Z3 STEP D
2580 X3=(H-B)/M
2590 SET(X3,H)
2600 NEXT H
2610 RETURN
2620 IF Y3>Z3 THEN O=-1 ELSE D=1
2630 FOR H=Y3 TO Z3 STEP O
2640 SET(X3,H)
2650 NEXT H
2660 RETURN
3000 REM SEASONS GREETINGS - ROUTINE 3
3010 CLS
3020 READ X,Y
3030 IFX=0 AND Y=0 THEN 3010
3040 PRINT@ X,CHR$(Y):GOTO 3020
3050 DATA 13,160,14,176,15,176,16,176,17,176,18,176
3060 DATA 72,160,73,184,74,156,75,143,76,131,77,131,78,
     131
3070 DATA 79,131,80,131,81,131,82,131,83,143,84,180
3080 DATA 136,139,137,143,138,189,139,176,140,144,147,1
     60
3090 DATA 148,176,149,176,150,176,154,176,155,176,156,1
     76
3100 DATA 157,176,158,176,159,144,161,176,162,176,163,1
     76
3110 DATA 168,176,169,176,170,176,171,176,175,176,177,1
     76
3120 DATA 178,176,179,176,183,176,184,176,185,176,186,1
     76
3130 DATA 187,144,196,176,197,152,198,148,199,180,200,1
     44
3140 DATA 203,138,204,131,205,139,206,173,207,180,208,1
     44
3150 DATA 209,160,210,191,211,141,212,148,213,148,214,1
     42
3160 DATA 215,129,216,184,217,159,218,129,220,160,221,1
     84
3170 DATA 222,143,224,139,225,173,226,176,227,146,228,1
     31
3180 DATA 230,188,231,151,232,129,235,186,236,149,238,1
     84
3190 DATA 239,159,240,131,242,184,243,159,244,129,246,1
     39
3200 DATA 247,173,248,176,249,144,250,138,251,131,252,1
     31
3210 DATA 258,168,259,183,260,144,269,168,270,184,271,1
     91
3220 DATA 272,133,273,138,274,143,275,148,276,148,277,1
     35
3230 DATA 278,129,279,138,288,139,281,148,282,148,283,1
     34
3240 DATA 284,143,285,133,287,131,288,148,289,148,298,1
     42
3250 DATA 291,135,294,139,295,141,296,148,297,148,298,1
     35
```

Program continues

Victor Andrews and Soft Sector Marketing, Inc.

It is time to put your word processing program away and use a word processing system

THIS COULD BE THE END OF YOUR LOVE AFFAIR?!!

Soft Sector Marketing
and
ABC Sales

Takes on Scriptset® By Radio Shack® and Electric Pencil©®

LAZY WRITER
© 1980 by DAVID WELSH

Easy to use, written all in machine code

HAS ALL THE THINGS THAT OTHER WORD PROCESSING PROGRAMS SHOULD HAVE

● It permits the inserting and deleting by characters, word, sentences, and paragraphs ● Page scrolling up and down ● Search ahead of the cursor or behind the cursor for any character ● The cursor can be moved up, down, left and right ● You can seek top of file and bottom of file ● Block move all text, block delete of text ● Search and replace or search delete ● Delete by character, word, sentence or paragraph ● Unlimited insert (to the limit of your machines memory) ● Permits use with lower case

AND IT HAS THINGS THAT OTHER PROGRAMS SHOULD HAVE BUT DON'T
Upper and lower case output to your printer (if your printer accepts lower case) without having your computer modified
ON UPPER CASE ONLY MACHINES This program marks the capital letters so you can see which letters are CAPITALS and which are not
● Will change all upper characters text to lower case or all lower case to upper. A SINGLE COMMAND ● Will capitalize the first letter of all sentences and all proper nouns ETC. WITH A SINGLE COMMAND ● LOADS ANY ELECTRIC PENCIL ● FILE ASCII SAVED FILES, EDTASM FILES OR BASIC PROGRAMS SAVED ASCII ● Permits installing special control characters in your text for your printers special features, like double wide or condensed print ● Definable screen length and definable print length to 255 characters wide ● Screen editing that is not final till your command. This means that you can edit your file on the screen and if you don't like how it reads you can cancel and leave it the way it was ● You can append files (which means that you can put one file to the end of another file) ● No lost characters at the end of the line even for the fastest typist ● A directory of all your files is available to the user with out leaving the program ● Saving programs to disk easy enough for the non-computer user ● To save memory, not all the program modules are in memory at one time but are called from the disk as needed ● you can set tabs positions like on a typewriter ● 10 CUSTOM COMMAND KEYS For the experienced user there is a command file that permits many special functions that are all user defined (not enough space for better explanation in ad, send for complete overview) ● Program has HELP file that is a short review of the commands that are available

STANDARD PRINTER MODULE
This printer module is provided for the user as a standard feature. Optional special printer routines for custom printer will be available in the near future. In this original release, it has the following printer drivers and will support the following printing devices. RS232, TRS232 AND PARALLEL printer ports. You have the following format commands. Justifies text, Centers text, Centers title, Line Spacing, Line Length from 3-255 characters and Set Margins. ● Also send any ASCII code to any printer from the text

● Save formated texted to the disk for spooling later ● Information for customer to load his own special printer driver ● Printing can be stopped and started by the user at any time and then restarted where you left off ● You can print entire file or just print to bottom of the page

COMMUNICATION PACKAGE
RS232 COMMUNICATION TERMINAL PROGRAM permits you to communicate with other computers, transfer files from one machine to another. Permits dumping memory across the phone lines. Receive files from other TRS-80's and "Shake Hands" with larger computers. This is the complete system called LAZY WRITER. There is no package written for the TRS-80 that is as comprehensive. This package is available for the TRS-80 mod 1 32k or larger with at least a single disk drive. List price is from $125.95. For a more complete overview send a self addressed envelope

SUPER-UTILITY
by K. WATT

SUPER-UTILITY was written by Kim Watt of Breeze Computing and is the most fantastic program of its kind to be available on the market at this time. SUPER-UTILITY is a machine language, stand alone program that has its own I/O routines and does not use any ROM or DOS calls. As a result, it should also operate on "CPM" machines and does not require that the disk be in any drive after initialization of the program. This program also supports the Radio Shack lower case modification, after initialization, SUPER-UTILITY occupies all memory from 4000H to 9FFFH. (That's 24K of machine code!!)

The zap utility is a program that does everything Apparat's "SuperZap" does, plus many additional enhancements. The zap utility allows the user to go to the heart of the disk and read and/or modify data regardless of whether it is a protected disk or not. The screen printout on zap is similar to SuperZap, (one sector in HEX and ASCII) but also tells user the true and relative tracks and whether sector is IBM format or not. Zap also has a search routine that locates and displays the highest or lowest track on the disk and another that locates and displays the highest or lowest sector on a given track. In addition, zap allows user to single step track to track or sector to sector and even allows user to output to the printer. Zap has dual cursors (one for ASCII and one for HEX) that move simultaneously and allows that user to modify data using hex, decimal, or ASCII input, and any changes are automatically updated on both sides of the readout. Zap also allows user to display disk sectors, file sectors, copy disk sectors, compare disk sectors, or display/modify main memory. In addition you may search memory or the disk for a specified string and have its location returned.

Purge is a utility that allows user to kill files by filespec or have the computer list them one at a time for deletion. In addition, purge will also zero out unused directory entries or zero out unused disk sectors. You may also compute passwords on files. In addition, you can kill files by naming the category of the files (example: /CMD/BAS/TXT, (I) invisible, (V) visible, etc.) and also may change the disk name, date, passwords, protection levels. Purge also contains a complete disk directory that indicates all active and non-active files on the disk and their location in the directory and the status of all granules on the disk.

Format is a utility that allows the user to format a disk with standard format, format WITHOUT ERASING existing data, or special format (custom format your disk any way you want it). This utility also allows the user to add tracks to any disk (example: change a 35 track disk to a 40).

The disk copy utility will copy any standard disk with or without formatting. The special disk copy allows the user to make a backup of "ANY" (that's right I said ANY) TRS-80* readable disk that is presently on the market, regardless of any efforts that have been made to protect the disk from being backed up. NOTE: (the only exception is that it won't copy itself). This program's only intended use is for user to make backups of his/her legally purchased programs for his/her own use. Please do not use this utility to make "bootleg copies" for others as authors of quality programs deserve to be paid for their royalties!

The tape copy utility allows the user to make backups of "ANY" TRS-80* readable tapes currently on the market regardless of any protection attempts or baud rate.

The disk repair utility allows the user to automatically repair the HIT and GAT sectors if damaged and will also automatically repair a damaged BOOT. This utility does a complete directory check and will advise user of any errors. In addition this utility allows the user to recover killed files (if the file was killed by this utility or by NEWDOS) and user may read protect the directory. This utility also advises you of all files that are on the disk the location of each and which are presently active.

The memory utility allows user to move memory, jump to memory, test memory, compare memory, zero memory, exchange memory, edit memory, load memory to/from disk, and input or output a byte to any port. Only $49.95 plus $2.50 shipping handling.

We Won't Waste Your Time

OPSYS 2™ is a multiple command processor and a potent job control language. Schedule entire systems of programs to run without your intervention.

OPSYS 2 simplifies your work. Changes the meaning of keys on your keyboard. A single stroke will invoke a sequence of commands. Create, save and retrieve libraries of keyboard programs.

OPSYS 2 does other smart things for you. We know your time is valuable.

- Over 30 modules for quick and easy disk and Basic operations
- Set time and date with minimum keystrokes
- Automatically executes chain files
- Write system tapes from DOS
- Read system tapes directly from DOS
- Display hex and decimal equivalents
- Maintain notes on screen during other operations
- Move memory blocks
- List memory with or without control codes
- Poke values into memory from DOS
- Peek values from memory from DOS
- Type ASCII text into memory
- Jump to any address
- Search/Replace Basic text
- Automatic command file using the DO command
- And many more

OPSYS 2 with complete documentation at the intelligent price of $79.95.

OPSYS 2™ is the Powerful TRS-80™ Model I operating system.

Use it with 35 or 40 track versions of TRSDOS,™ NEWDOS,™ or NEWDOS+.™

↙21

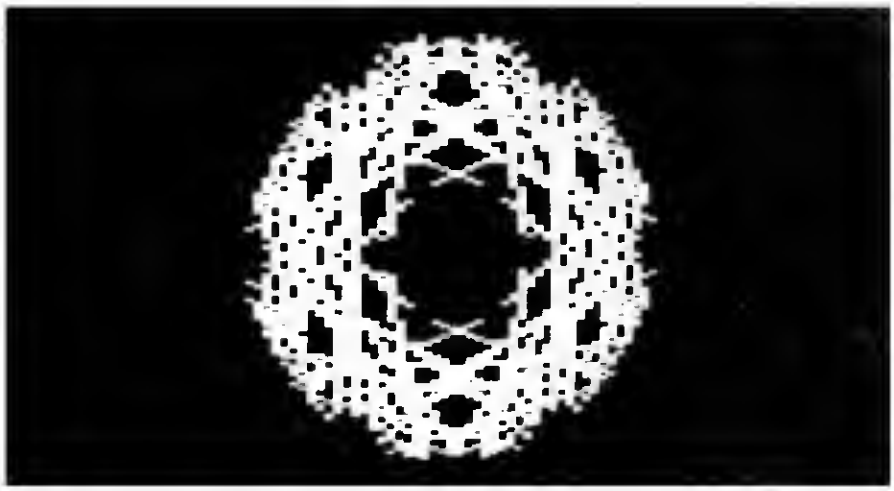
PROGRAMMA INTERNATIONAL INCORPORATED

3400 Wilshire Boulevard
Los Angeles, California 90010
(213) 384-1116

TRSDOS trademark of Tandy, Inc.
NEWDOS and NEWDOS+ trademark of APPARAT

```
3260 DATA 299,129,300,136,301,142,302,131,304,136,305,1
     90,306,179,313,107,314,159
3270  DATA 323,130,324,131,325,143,326,140,327,100,320,
     100
3280 DATA329,100,330,100,331,100,332,143,333,135,334,13
     1,341,176,342,176,343,152,344,140
3290 DATA 345,140,346,140,347,140,340,140,349,140,350,1
     40
3300 DATA 351,140,352,140,353,140,354,140,355,140,356,1
     40
3310 DATA 357,140,350,140,359,140,360,140,361,140,362,1
     40
3320 DATA 363,140,364,140,365,140,366,140,367,140,360,1
     40
3330 DATA 369,140,370,140,371,140,372,140,373,140,374,1
     40,375,140,376,135,377,131,306,160
3340 DATA 404,175,405,176,406,144,423,160
3350 DATA 424,104,425,132,429,160,433,160,434,176,435,1
     40
3360 DATA 436,140,437,129,451,131,452,172,453,100,454,1
     00
3370 DATA 455,140,456,143,457,131,450,131,459,131,460,1
     75
3380 DATA 461,191,470,131,471,131,472,141,473,140,474,1
     40
3390 DATA 475,140,476,140,477,140,470,140,479,140,480,1
     40
3400 DATA 401,140,402,140,403,140,404,140,405,140,406,1
     90
3410 DATA 487,159,400,141,409,140,490,132,491,136,492,1
     43
3420 DATA 493,129,494,140,495,140,496,131,497,131,514,1
     76
3430 DATA 515,190,516,143,517,129,519,131,520,131,521,1
     31
3440 DATA 522,131,523,131,524,131,527,187,528,157,529,1
     34
3450 DATA 530,131,531,139,532,132,533,160,534,100,535,1
     79
3460 DATA 536,179,537,107,530,132,540,160,541,100,542,1
     79
3470 DATA 543,179,544,179,545,157,548,160,549,198,550,1
     35
3480 DATA 553,160,554,190,555,135,557,104,550,159,559,1
     34,560,131
3490 DATA 561,131,561,171,562,109,564,160,565,100,566,1
     35
3500 DATA 567,163,560,191,569,151,570,160,571,103,572,1
     79
3510 DATA 573,131,574,139,575,132,577,191,578,191,502,1
     76
3520 DATA 503,140,504,134,505,131,506,131,507,131,500,1
     88
3530 DATA 509,160,590,191,591,135,597,175,598,101,599,1
     76
3540 DATA 600,104,601,140,604,175,605,181,606,176,607,1
     04,600,140
3550 DATA 611,104,612,159,613,129,616,104,617,159,610,1
     29,619,176,620,190,621,135,623,160,624,104
3560 DATA 625,143,626,129,627,136,620,191,629,177
3570 DATA 630,172,631,191,632,135,636,130,637,191,630,1
     40
3580 DATA 641,191,642,191,646,130,647,131,648,140,649,1
     40
3590 DATA 650,140,651,190,652,159,653,140,654,140,655,1
     40
3600 DATA 656,140,657,140,658,100,659,144,660,160,669,1
     44
3610 DATA 675,139,676,141,677,140,678,140,600,131,609,1
     31
3620 DATA 693,104,694,159,695,129,699,176,700,150,701,1
     35
3630 DATA 705,130,706,139,707,173,700,100,709,176,710,1
     76
3640 DATA 711,176,712,176,713,104,714,150,715,135,716,1
     29
3650 DATA 720,172,721,144,723,162,724,107,725,100,726,1
     76,727,176
3660 DATA 720,176,729,176,730,176,731,176,732,176,733,1
     78
```

Program continues



```
3670 DATA 734,185,735,180,736,176,737,176,738,176,739,1
     76
3680 DATA 740,176,741,176,742,176,743,176,744,176,745,1
     76
3690 DATA 746,176,747,176,748,176,749,176,750,176,751,1
     76
3700 DATA 752,176,753,176,754,176,755,176,756,190,757,1
     83
3710 DATA 758,176,759,140,760,140,761,134,762,131,780,1
     76
3720 DATA 781,140,782,140,783,134,784,131,785,139,786,1
     67
3730 DATA 787,177,788,176,789,143,796,160,797,152,800,1
     31
3740 DATA 801,137,802,140,803,140,804,164,805,176,806,1
     76
3750 DATA 807,176,808,176,809,176,810,176,811,176,812,1
     76
3760 DATA 813,176,814,176,815,152,816,140,817,140,818,1
     31
3770 DATA 819,129,842,136,843,191,844,145,856,176,857,1
     76
3780 DATA 858,184,859,142,860,131,908,130,909,131,910,1
     37
3790 DATA 911,140,912,140,913,140,914,140,915,140,916,1
     40
3800 DATA 917,140,918,131,919,131,0,0
3810 FOR X=1 TO 3200:NEXTX
4000 REM POEM SEQUENCE - ROUTINE 4
4010 CLS:PRINTCHR$(23):PRINT"STOPPING BY WOODS"
4020 PRINT"      ON A SNOWY EVENING":PRINT
4030 PRINT"            BY ROBERT FROST"
4040 FOR X=1TO2000:NEXTX:CLS
4050 FORZ=1TO150
4060 SET(RND(127),RND(47)):FORR=1TO30:NEXTR:NEXTZ
4070 PRINT@460,CHR$(230);:GOSUB4440 :PRINT@525,CHR$(230
     );:I=0:GOSUB4440
4080 PRINT@525," WHOSE WOODS THESE ARE I THINK I KNOW."
     ;
4090 GOSUB4440
4100 PRINT@525," HIS HOUSE IS IN THE VILLAGE, THOUGH; "
     ;
4110 GOSUB4440
4120 PRINT@525," HE WILL NOT SEE ME STOPPING HERE     "
     ;
4130 GOSUB4440
4140 PRINT@525," TO WATCH HIS WOODS FILL UP WITH SNOW
     ";
4150 GOSUB4440
4160 PRINT@525," MY LITTLE HORSE MUST THINK IT QUEER
     ";
4170 GOSUB4440
4180 PRINT@525," TO STOP WITHOUT A FARMHOUSE NEAR     "
     ;
4190 GOSUB4440
4200 PRINT@525," BETWEEN THE WOODS AND FROZEN LAKE   ";
4210 GOSUB4440
```

Program continues

NEW DISC DRIVES FOR TRS—80* "FLIPPY"—250K bytes/disc FAST—5ms Track to Track Access

REAL VALUE

AEROCOMP offers the best value in microcomputer disc drives on the market today! Reliability, features and cost tough to beat. We deliver...and we stand behind our products, as evidenced by the only FREE TRIAL OFFER in the industry. Examine your systems needs and order today!

MYSTERY REMOVED

There appears to be some confusion in the terminology used to describe disc drives and their features. Here's what we mean:

***FLIPPY** Allows the use of both sides of a diskette with a single-headed drive by simply turning the diskette over (Model 40-1).

***TRACK DENSITY** specified in tracks per inch (TPI). Refers to the number of tracks per radial inch on the diskette. Typically 48 TPI=40 usable tracks and 96 TPI=80 useable tracks.

***DOUBLE DENSITY** refers to recording density in bits per inch (bpi). Typically single density means data can be recorded up to 2,938 bpi; double density means data can be recorded up to 5,876 bpi.

***DOUBLE-SIDED** refers to number of read/write heads. Single-sided is one head, read/write one side only; double-sided is dual heads allowing read/write operations on both sides of the diskette. A double sided drive appears as two separate drives to the controller.

***CAPACITY** unformated capacity is the total amount of storage space available on a diskette. Typically 125K bytes on a 40 track 5.25in. diskette. Formated capacity is the total USABLE storage space on a diskette. Typically 102K bytes on a 40 track 5.25in. diskette.

***ACCESS TIME** the time required for the head to move from one track to the next. Typically 5 to 40 milliseconds (ms).

FREE TRIAL OFFER

Order your AEROCOMP Disc Drive and use it with your system for up to 14 days. If you are not satisfied for ANY REASON (except misuse or improper handling), return it, packed in the original shipping container, for a FULL REFUND. We have complete confidence in our products and we know you will be satisfied! ORDER TODAY!

WARRANTY

We offer you a 90 day unconditional warranty on parts and labor against any defect in materials and workmanship. In the event service, for any reason, becomes necessary, our service department is fast, friendly and cooperative.

100% TESTED

AEROCOMP Disc Drives are completely assembled at the factory and ready to plug in when you receive them. Each drive is 100% bench tested prior to shipment. We even enclose a copy of the test checklist, signed by the test technician, with every drive. AEROCOMP MEANS RELIABILITY!

ORDER NOW

● MODEL 40-1 DISC DRIVE $339.95ea. Single-sided, "Flippy", 48TPI. (40 track: single density unformated 125K bytes/side; double density unformated 250K bytes/side).

● MODEL 80-2 DISC DRIVE $439.95ea. Double-sided, 48TPI. (80 track/40 per side; single density unformated 250K bytes; double density unformated 500K bytes).

● MODEL 80-1 DISC DRIVE $439.95ea. Single-sided, "Flippy", 96TPI. (80 track: single density unformated 250K bytes/side; double density unformated 500K bytes/side).

● MODEL 160-2 DISC DRIVE $595.95ea. Double-sided, 96TPI. (160 track/80 per side: single density unformated 500K bytes; double density unformated 1 megabyte).

All models are capable of single or double density and are complete with power supply and silver enclosure. Send for information on AEROCOMP 2- and 3-drive systems available in 40 and 80 track.

● SELECT EITHER A 2-DRIVE OR 4-DRIVE CABLE FOR USE WITH YOUR DRIVE(S):
2-DRIVE CABLE (for use with 1-or 2-drive systems) **$24.95ea.**
4-DRIVE CABLE (for use with 1-, 2-, 3-or 4-drive systems) **$34.95ea**
Add $1.25 shipping and handling

● MINI DISKETTES (5.25 in). Box of 10 **$39.95** add $1.25 shipping and handling

● DISC OPERATING SYSTEMS

NEWDOS+(40 TRACK)	$109.00
NEWDOS/80 (80 TRACK)	$149.00

★ SPECIAL COMBINATION OFFER ★

● Model 40-1 Disc Drive	$339.95
● 2 Drive Cable	24.95
● Disc Operating System (NEWDOS+)	109.00
● Freight	5.25
Reg. $479.15	
Special $399.95	

SPECIAL COMBO EXCEPT WITH NEWDOS/80
Reg. $519.15 **Special $439.95**

To order by mail, specify Model Number(s) of Drive, cable, etc. (above), enclose check, money order, VISA or MASTERCHARGE card number and expiration date, or request COD shipment. Texas residents add 5% sales tax. ADD $4.00 per drive for shipping and handling. Please allow 2 weeks for personal checks to clear our bank. No personal checks will be accepted on COD shipments-cash, money orders or certified checks only. You will receive a card showing the exact COD amount before your shipment arrives. Be sure to include your name and shipping address. WE SHIP PROMPTLY! In the event there is a slight delay, you will be notified of the shipping date and we will NOT deposit your money order or charge your bankcard until the day we ship!

COMPARE AND BUY AEROCOMP!

	"FLIPPY"	ACCESS TIME (track to track)	HEAD LOAD SOLENOID	DISC EJECTOR	CAPACITY (unformated single density)	EASY-ENTRY DOOR	FREE TRIAL
AEROCOMP	YES	5ms.	YES	YES	250K bytes (both sides)	YES	YES
RADIO SHACK*	NO	40ms	YES	NO	109K bytes	NO	NO
PERCOM	YES	25ms	YES	NO	250K bytes (both sides)	YES	NO
MPI	NO	5ms	YES	YES	125K bytes	YES	NO
SHUGART	NO	40ms	YES	NO	109K bytes	NO	NO
SIEMENS	NO	25ms.	YES	NO	125K bytes	YES	NO
TANDON	NO	5ms.	NO	NO	125K bytes	NO	NO
PERTEC	YES	25ms	YES	NO	250K bytes (both sides)	NO	NO
BASF	NO	12ms	YES	NO	125K bytes	NO	NO

Factual material from current manufacturer's data sheets is believed reliable but cannot be guaranteed comparing Aerocomp Model 40-1 to similar models

The TRS 80* expansion interface limits the track to track access time to 12ms *Trademark of Tandy Radio Shack

CALL TOLL FREE FOR FAST SERVICE (800) 824-7888, OPERATOR 24
FOR VISA/MASTERCHARGE/C.O.D. ORDERS
California dial (800) 852-7777, Operator 24. Alaska and Hawaii dial (800) 824-7919, Operator 24.
TOLL FREE LINES WILL ACCEPT ORDERS ONLY!
For Applications and Technical information, call (214) 337-4346 or drop us a card.

Dealers inqiries invited ↙387

AEROCOMP
Redbird Airport, Bldg. 8
P.O. Box 24829
Dallas, TX 75224

↙ Reader Service—see page 226

```
4220 PRINT@525," THE DARKEST EVENING OF THE YEAR    ";
4230 GOSUB4440
4240 PRINT@525," HE GIVES HIS HARNESS BELLS A SHAKE  ";
4250 GOSUB4440
4260 PRINT@525," TO ASK IF THERE IS SOME MISTAKE.    ";
4270 GOSUB4440
4280 PRINT@525," THE ONLY OTHER SOUND'S THE SWEEP  ";
4290 GOSUB4440
4300 PRINT@525," OF EASY WIND AND DOWNY FLAKE.    ";
4310 GOSUB4440
4320 PRINT@525," THE WOODS ARE LOVELY, DARK, AND DEEP,"
     ;
4330 GOSUB4440
4340 PRINT@589," BUT I HAVE PROMISES TO KEEP,  ";
4350 I=3:GOSUB4440
4360 PRINT@653," AND MILES TO GO BEFORE I SLEEP,   ";
4370 I=6:GOSUB4440
4380 PRINT@717," AND MILES TO GO BEFORE I SLEEP.   ";
4390 I=9:GOSUB4440
4400 FOR A=1TO800
4410 SET(RND(127),RND(47)):NEXTA
4420 CLS:PRINTCHR$(23):PRINT@76,"*";:PRINT@170,"*";:PRI
     NT@322,"BUT I HAVE PROMISES TO KEEP,";:PRINT@512,"
     AND MILES TO GO BEFORE I SLEEP..";
4430 PRINT@644,"*";:PRINT@690,"*";:PRINT@850,"*";:GOTO5
     010
4440 FORN=1TO20
4450 X=RND(127):Y=RND(47)
4460 IF Y=24+I GOTO4450
4470 IF Y=25+I GOTO4450
4480 IF Y=26+I GOTO4450
4490 SET(X,Y)
4500 FORA=1TO40:NEXTA
4510 NEXTN
4520 RETURN
5000 REM SNOW SCENE - ROUTINE 5
5010 FORL=1TO16
5020 A$(L)="":READ N
5030 FOR Z=1TON
5040 READY:A$(L)=A$(L)+CHR$(Y)
5050 NEXTZ:NEXTL
5060 PRINT CHR$(20)
5070 FORL=1TO16
5080 PRINTA$(L);
5090 NEXTL
5100 FORY=45TO47:FORX=122TO127
5110 SET(X,Y):NEXTX:NEXTY
5120 FORN=1TO2000:NEXTN
5130 FORN=1TO21
5140 READP,Q:S(N,0)=P:S(N,1)=Q:NEXTN
5150 FORR=1TO10
5160 FORN=1TO21
5170 P=S(N,0):Q=S(N,1):RESET(P,Q):NEXTN
5180 FORT=1TO21
5190 P=S(T,0):Q=S(T,1):SET(P,Q):NEXTT
5200 NEXTR
5210 GOTO5780
5220 DATA 9,197,144,215,136,209,172,198,144,202
5230 DATA 19,200,160,184,180,191,191,191,191,189,188,17
     6,215
5240 DATA 129,195,160,186,189,176,207
5250 DATA 25,200,191,191,191,191,191,191,191,191,191,19
     1,149
5260 DATA 207,129,201,176,190,191,191,191,141,176,199,1
     30,197
5270 DATA 27,194,129,197,130,139,143,191,191,191,191,15
     9,143
5280 DATA 131,197,130,211,135,128,176,191,191,191,189,1
     88,180,144,203
5290 DATA 27,196,144,206,160,203,129,199,160,196,188,19
     1,191
5300 DATA 191,191,191,191,191,191,143,188,188,180,176,1
     44,194,160,194
5310 DATA 30,194,160,186,190,129,199,168,191,176,198,14
     4,197
5320 DATA 190,176,188,188,191,191,191,191,191,191,191,1
     91,189
5330 DATA 176,196,130,133,197
5340 DATA 43,120,160,184,190,183,179,202,176,188,191,19
     1,191
5350 DATA 191,188,176,200,186,176,195,168,143,131,131,1
     31,131
```

Program continues

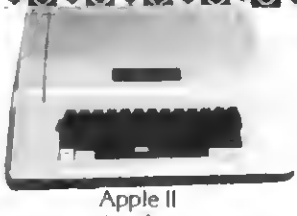
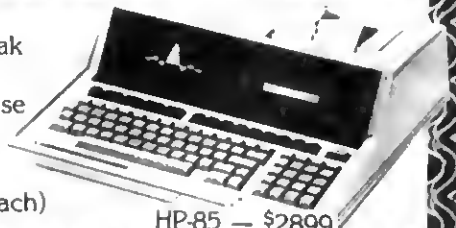
ΩMEGA WHOLESALE COMPUTER PRICES

SALES CO. ⌐389

DIRECT TO THE PUBLIC

12 Meeting St., Cumberland, R.I. 02864

Christmas Specials

When you buy:	You receive:	You receive:	When you buy:
 Atari 800 $749	1-8K plug in RAM Module FREE ($124.95 value)	1 Box (10 Pcs) 8" Diskettes FREE	 TRS-80 Model II - $3449
 Apple II 16K - $999 (Regular or Plus)	1 Apple Tape Recorder FREE ($40 value)	Microsoft Basic for only $162.50	 INTERTEC SUPERBRAIN 32K RAM - $2449.00 64K RAM - $2649.00
 NEW! ATARI 825 PRINTER — $949	Atari Interface Module FREE ($219.95 value)	Statistics Pak or Carrying case FREE ($95 value each)	 HP-85 — $2899
APPLE II DISK with Controller — $495	FREE 1 Box (5¼ Pcs) Diskettes	EPSON MX-80 80 Character, 9x9 Dot Matrix Roll & Pin Feed Printer for your Apple, TRS-80 or Commodore $599	
NEC Spinwriter 5530 or 5510 $2449	SOROC IQ 120 $699	FREE Interface to Your Computer	
OKIDATA MICROLINE 80 — $529			

VISA

master charge

ΩMEGA OFFERS THE BEST DELIVERY AND PRICE ON:
APPLE • ATARI • TRS-80 MODEL II • INTERTEC •
T.I. 810 • HEWLETT-PACKARD-85 • SOROC •
COMMODORE • NEC • QUME • CENTRONICS

ΩMEGA sells only factory fresh, top quality merchandise to out customers.
ΩMEGA will try to match any current advertised price with similar purchase conditions.
Before you buy anywhere else — be sure to call ΩMEGA Sales Co.

1-401-722-1027

ΩMEGA ships via UPS, truck or air. COD's, VISA, Mastercharge accepted.
"A member in good standing of the Better Business Bureau."

We carry a complete line of the above equipment. For information and further pricing call:

TOLL FREE
1-800-556-7586
TELEX: 952106

Reader Service—see page 274

```
5360 DATA 176,187,191,191,191,191,191,191,191,191,191,1
     91,188
5370 DATA 188,188,176,176,196
5380 DATA 58,128,129,184,191,191,191,191,188,176,195,17
     6,188
5390 DATA 143,131,175,191,191,191,191,188,176,178,131,1
     35,194
5400 DATA 160,140,172,191,191,191,189,176,144,128,176,1
     76,190
5410 DATA 191,135,191,191,143,191,191,191,191,191,191,1
     35,131
5420 DATA 131,143,131,131,129,138,196
5430 DATA 60,140,175,191,191,191,191,172,180,176,129,19
     4,130
5440 DATA 128,184,190,191,175,191,191,191,191,178,179,1
     79,139
5450 DATA 128,160,184,188,191,191,191,131,143,140,140,1
     42,179
5460 DATA 179,179,179,188,140,140,128,172,179,143,191,1
     91,191
5470 DATA 191,189,180,180,196,160,180,190,191
5480 DATA 64,134,187,191,191,191,191,189,180,178,139,13
     2,176
5490 DATA 188,183,179,180,188,191,191,191,191,191,191,1
     91,191
5500 DATA 189,191,189,188,191,143,179,188,128,143,143,1
     43,179
5510 DATA 179,179,188,188,188,191,191,140,143,143,143,1
     43,131,131,131,131,131,131,131,179
5520 DATA 179,188,191,191,143,191
5530 DATA 54,191,191,191,191,191,191,191,191,191,191,19
     1,191
5540 DATA 191,191,191,191,191,191,191,191,191,191,191,1
     91,143
5550 DATA 191,143,179,188,188,143,143,179,188,191,143,1
     43,131
5560 DATA 131,131,203,176,180,191,191,191,191,191,159,1
     43,179
5570 DATA 188,128,188
5580 DATA 58,191,191,191,191,191,191,191,191,191,191,19
     1,191
5590 DATA 191,191,143,191,191,143,143,143,179,179,179,1
     88,128
5600 DATA 188,191,143,179,188,191,143,131,207,176,190,1
     91,191
5610 DATA 191,143,131,143,179,188,190,191,143,179,128,1
     79
5620 DATA 49,191,191,143,191,191,191,143,143,143,179,17
     9,179
5630 DATA 188,188,128,188,188,191,143,143,143,179,179,1
     79,128
5640 DATA 179,188,191,191,143,129,208,138,191,191,191,1
     91,191
5650 DATA 191,128,182,131,135,137,188,191,191,191,191
5660 DATA 45,179,179,128,188,188,188,191,191,143,143,14
     3,179
5670 DATA 179,179,128,180,188,180,191,191,191,191,191,1
     91,191
5680 DATA 191,143,135,129,212,131,191,191,191,188,176,1
     52,143
5690 DATA 175,191,140,132,129,162,191
5700 DATA 41,143,143,128,179,179,180,180,188,191,191,19
     1,191
5710 DATA 191,191,191,191,191,191,191,143,143,143,131,1
     31,216
5720 DATA 160,184,191,191,131,131,191,143,135,179,180,1
     82,139
5730 DATA 144,162,191
5740 DATA 32,191,191,188,159,143,143,143,143,143,131,13
     1,131
5750 DATA 131,131,129,221,176,176,188,188,191,191,191,1
     88,194
5760 DATA 152,143,143,167,179,188,191
5770 DATA 125,14,86,37,70,6,22,15,82,31,49,9,89,43,10,2
     ,72,35,64,12,42,46,108,2,6,9,63,40,81,14,50,45,56,
     17,119,6,92,33,84,3,59,1
5780 FDRA=1TO2000
5790 X=(RND(128))-1:Y=(RND(40))-1
5800 SET(X,Y)
5810 X=RND(127):Y=(RND(23))+24:SET(X,Y):NEXTA
5820 FDR Y=47 TO 8 STEP -1
```

Program continues

SAFOR

Sales Analyst and Forecaster

with Graphic Display

Here is a new program from Software Etc. . . that is invaluable to any businessman. SAFOR is a time series analysis and forecasting program that will produce presentation quality graphs on your own printer. SAFOR will handle up to ten years of monthly data, provide a comprehensive analysis of past patterns and make both long and short-range forecasts.

SAFOR uses a classical time series decomposition model to provide both tabular and graphic presentations of Original Data, 12 Month Forecasts, Business Cycle, Growth Cycle, Seasonal Pattern and 12 Month Moving Average.

SAFOR can handle any type of data series measured on a monthly basis where no value is zero or less. This data may be in any unit of measure, however SAFOR does not adjust dollar figures for inflation. The trend is calculated in the same units as actual sales, while other factors are treated as multipliers or indexes. Because the irregular component is unpredictable, SAFOR ignores it.

SAFOR is designed for ease of use, data entry is particularly straightforward, every user action is prompted. The program features extensive editing, updating, data storage and error detection routines.

SAFOR allnws for varying levels of expertise in the techniques of analysis, beginner to expert. For the beginner, SAFOR contains standard default operations to help prepare routine analysis and forecasts. For the expert, key assumptions in the program are readily modified without any programming.

SAFOR will run on your TRS-80® Level II with an expansion interface, 32K of RAM memory, a disk drive, TRSDOS® and optionally a 132 column printer.

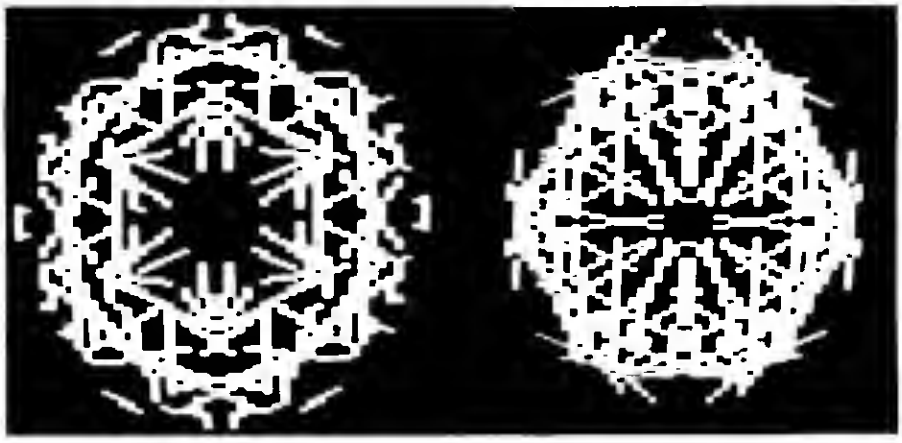
Each program comes complete with a two year set of demonstration data. Order yours now! A $200.00 value at this Introductory Price of only **$79.95** Good through December 31, 1980 Only.

✔42

Software Etc. . .
1839 Chamberlain Drive,
Carrollton, Texas 75007.
Phone Orders: (214) 492-0515

Demand a Demonstration from your local dealer, or write for a brochure of our complete line of fine software.

```
5030 FOR X=0 TO 127
5040 SET (X,Y)
5050 T=RND(127):W=Y-2
5060 IFW<0 THEN 5900
5070 SET(T,W)
5080 T=RND(127):W=RND(47)
5090 SET(T,W)
5900 NEXTX:NEXTY
5910 FORX=1TO500:NEXTX
6000 REM SIGNATURE PAGE - ROUTINE 6
6010 CLS:P$="*":PRINT CHR$(23)
6020 L=LEN(B$):L=INT(L/2)*2
6030 S1$=STRING$(6,"*")+CHR$(212)+STRING$(6,"*")
6040 FOR J=0 TO 950 STEP 10
6050 PRINT@J,P$:NEXT J
6060 FOR J=1 TO 75:NEXT J
6070 FOR K=0 TO 190 STEP 10:PRINT@X,P$:NEXT K
6080 PRINT@192,S1$
6090 PRINT@256,P$
6100 PRINT@200,"H A P P Y"
6110 PRINT@318,P$:PRINT@320,P$:PRINT@302,P$:PRINT@304,P
     $
6120 PRINT@402,"H O L I D A Y S"
6130 PRINT@446,P$:PRINT@448,P$:PRINT@510,P$:PRINT@512,P
     $
6140 PRINT@540,"FROM"
6150 PRINT@574,P$:PRINT@576,P$:PRINT@638,P$:PRINT@640,P
     $
6160 PRINT@(672-L),B$
6170 PRINT@702,P$
6180 PRINT@704,S1$
6190 FOR K=770 TO 950 STEP 10
6200 PRINT@K,P$:NEXT K
6210 FOR J=1 TO 2500:NEXT J
7000 RESTORE: REM ROUTINES 3 & 5
7010 GOTO 2000 :  REM REPEAT ROUTINES 2 THRU 5
```

Program Listing

```
10 RANDOM:CLS:PRINT:PRINT
20 PRINT"SNOWFLAKES"
30 PRINT"COPYRIGHT 1980 BY"
40 PRINT"VALERIE VANN"
50 PRINT"631 G ST., DAVIS, CA."
60 PRINT:PRINT"LEVEL II BASIC"
70 FOR J=1 TO 800:NEXT J
130 REM TITLE PAGE
140 CLS
150 PRINT CHR$(23)
160 FOR J=2 TO 442 STEP 0
170  PRINT@J,"*"
180 FOR F=1TO10:NEXT F
190 NEXT J
200 PRINT@450,"* *  S N O W F L A K E S  * *"
```

Program continues

Toll-free order no.:
1-800-527-4196

LOOK!!

TRS-80*Owners Save on Equipment & Software!

ACCESS Mini-disk Systems

Access Unlimited's own economy mini-disk systems store more data, are more reliable. Data access times are fastest possible with your Expansion Interface. Heavy duty power supplies run cooler, last longer. Low noise three-wire ac power cord is safer. Enclosures are finished in compatible silver enamel.

AFD-100' (40-track, 102 Kbytes) $315.00

Mention our DOUBLE DISCOUNT NUMBER when you order and save $20.00 on your AFD-100!

Percom Mini-disk Systems

TFD-100™ (40-track, 102 Kbytes/side) $349.95
TFD-200™ (77-track, 197 Kbytes) $634.95
Prices include Percom upgrade PATCHPAK™

DATA SEPARATOR™

This PC board plug-in adapter for the TRS-80* virtually eliminates data read errors (CRC error — Track locked out!) which occur on high density inner disk tracks, a problem that has plagued TRS 80* systems. The Percom Data Separator™ is installed in the Expansion Interface without modifying the host system. Caution: Opening the TRS-80* Expansion Interface may void the limited 90-day warranty: $29.95.

Percom OS-80™

An advanced easy-to-use disk operating system that works with Level II BASIC commands. Resides in only 7-Kbyte of memory. May be extended indefinitely with disk-resident utilities. Supplied on 5" disk with example programs: $29.95 with instructions.

CIRCLE J Software

Two extremely useful utilities for Percom's OS-80™ DOS:

1. Machine Language Save/Load Utility. On 5" disk with bonus patch program that allows RS Renumber Utility to run under OS-80™. $14.95, with instructions.
2. VARKEEP — Adds NAME SAVE and NAME KEEP commands to OS-80™. Use one set of common data with two or more BASIC programs. Also runs under Radio Shack DOS. On 5" disk, with instructions: $14.95.

Z80ZAP

Super fast machine language disk modification utility. Read, Write, Display, and Modify sectors; remove passwords; apply patches, fixes; make backups and much more. On 5" disk with instructions: $29.95.

Ask about Scott Adams' Adventure games!

100% machine language word processor . . .
SPECIAL DELIVERY (From Softwsre Etc.)

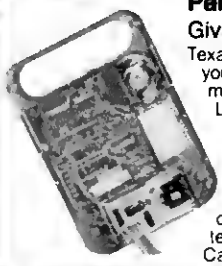
Use MAILFORM to create name and address lists; EXTRACT to find names by ZIP, address, gender, age, etc.; SORT to sort an entire list on any field in seconds. Print personalized letters written with either the Electric Pencil‡ or Scripsit* using MAILRITE. Prints labels from Mailfile created under MAILFORM. Runs under Percom's OS-80™, Radio Shack's TRSDOS*. $125 (disk)

Inexpensive Color Graphics: the Percom Electric Crayon™

Spectacular multicolor graphics, sharp 2-color alphanumerics with your TRS-80*, a color tv and the Percom Electric Crayon™. Up to eight colors. Resolution with full display memory (6 Kbytes) is 256 X 192 picture elements. Microprocessor controlled (the Electric Crayon™ is not only a full color graphics system but also a complete, self-contained control computer with a dual bidirectional parallel I/O port — provision for second dual port. Interface the TRS-80* via your Expansion Interface or Printer Cable Adapter. Supplied with 1 Kbyte display RAM, EGOS™ operating system and comprehensive users manual with example programs. $249.95.

Optional TRS-80* interconnecting cable: $24.95.

Percom's Speak-2-Me-2™
Give your TRS-80* the gift of speech



Texas Instruments' Speak & Spell* is the voice of your TRS-80* computer with this clever interface module manufactured by Percom. Your own Level II BASIC programs announce, command, implore with sentences and expressions formed from Speak & Spell's* vocabulary. The Speak-2-Me-2™ PC module installs in the battery compartment of your Speak & Spell*. Power is supplied from an ordinary calculator power pak. Comes with interconnecting cable (for TRS-80* EI or Printer Cable Adapter), operating software and users manual: $69.95. (Speak & Spell* not included.)

the DOUBLER™

Percom's new plug-in adapter for your Expansion Interface stores almost twice the data on a diskette track as a single-density system. You can store up to four times more data — depending on the type of drive — on one side of a diskette than you can store using a standard Model I mini-disk drive. Other features: Reads, writes and formats *either* single or double density minidiskettes. • Runs TRSDOS*, NEWDOS +, Percom OS-80™ or other single density software *without* changing either software or hardware. Switch to double-density when convenient. • Includes DBLDOS™ a TRSDOS* compatible double-density operating system. • Includes on card, high-performance data separator circuit. • Installs *without* rewiring or trace cutting. • Introductory price, including DBLDOS and format conversion utility, only $219.95. Mention our DOUBLE DISCOUNT NUMBER when you order and save $20.00 on your DOUBLER!

DOUBLE-ZAP — Modifies NEWDOS 80 + for double-density operation using the Percom DOUBLER. Permits simultaneous single- or double-density operation. From Software Etc. and Circle J Software. On minidiskette. $49.95.

Use your credit card and save!
VISA and MasterCard charges are not deposited until the day your order is shipped.

Disk System Interconnecting Cables

Improvement over RS cable design places drive 0, which includes the cable termination, at the end of the cable to eliminate the reflected noise of an unterminated cable. Better data integrity. Prices:

Two-Drive Cable $24.95
Four-Drive Cable 34.95

Power Line Filter

115/250 V, 50-400 Hz. Instructions included for easy installation in standard mini-box chassis. $19.95

Minidiskettes

10 Disks in a convenient plastic organizer box . . . $34.90
Single Disk 3.49

Disk Drive ID Tabs

1" x 1-¼" self-adhering plastic drive identification tabs. Compatible silver with engraved black drive number. Two tabs (Nos. 0, 1): $2.50; three tabs (Nos. 0, 1, 2): $3.25, four tabs (0, 1, 2, 3): $4.50.

How to Order

Order by calling Access Unlimited toll-free on 1-800-527-4196†.

Mail orders also accepted. Orders may be charged to a VISA or Master Charge account, or paid by a cashier's check, certified check or money order. We accept COD orders with 25% deposit. Sorry, we cannot accept personal checks. We pay shipping and insurance charges on orders over $1,000.00. Add approximate insurance and shipping charges for under $1,000.00. If in doubt about these charges, ask when you call in your order. Texas residents include 5% sales tax. Minimum order $20.00. Allow 2 to 4 weeks for delivery. **†Texas residents call (214) 494-0206.**

ALL PRICES AND SPECIFICATIONS SUBJECT TO CHANGE AND ALL OFFERS SUBJECT TO WITHDRAWAL WITHOUT NOTICE.

DOUBLE DISCOUNT NUMBER:
80M110

ACCESS UNLIMITED
315 N. Shiloh · Ste. D1 · Garland, TX 75042
(214) 494-0206

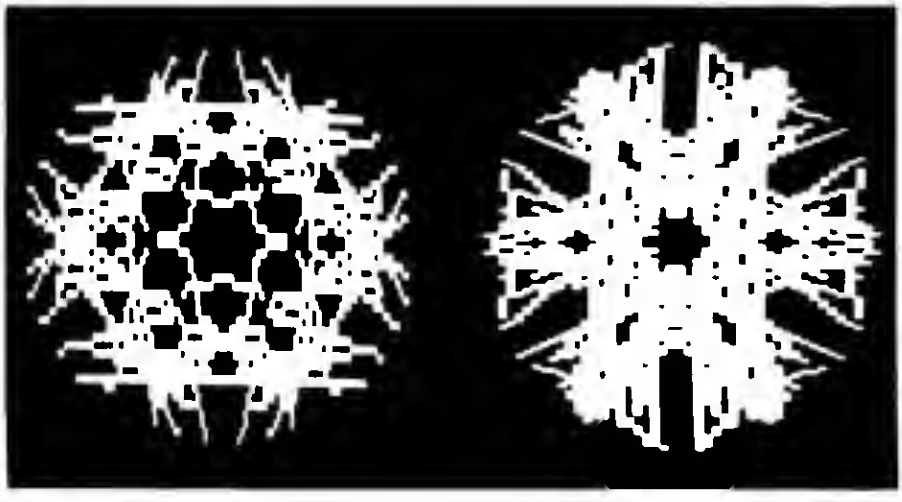
† trademark of Apparat Corporation
‡ trademark of Access Unlimited Company
™ trademark of Percom Data Company, Inc.
* RADIO SHACK and TRS-80 are trademarks of Tandy Corporation
* trademark of Texas Instruments Corporation
‡ trademark of Michael Shrayer Software, Inc.

```
210 FOR J=514 TO 958 STEP 0
220  PRINT@J,"*"
230  FOR F=1 TO 10 :NEXT F
240 NEXT J
250 FOR J=1 TO 950:NEXT J
260 REM INITIATE & SET NO. OF REPEATS
270 V=120/40
280 P=3.141592654
290 FOR E=1 TO 5
300 CLS
310 REM DEFINE LINE & NO. OF LINES
320 FOR X=1 TO (RND(5)+7)
330   X=RND(24)
340   Y=RND(24)
350   R=SQR(X[2+Y[2)
360   IF R>24 THEN 330
370   T=RND(24)
380   Z=RND(24)
390  S=SQR(T[2+Z[2)
400  IF S>24 THEN 370
410  GOSUB 520
420 REM MIRROR IMAGE OF LINE
430  Y=-1*Y
440  Z=-1*Z
450  GOSUB 520
460 REM DELAY LOOP IN LEVEL 3 VERSION
470 NEXT K
480 FOR I=1 TO 2000:NEXT I
490 NEXT E
500 GOTO 140
510 REM SUBROUTINE-PLOT & ROTATE
520 W=ATN(Y/X)
530 Q=ATN(Z/T)
540 FOR J=1 TO 6
550   W=W+P/3
560   Q=Q+P/3
570   X2=R*COS(W)
580   Y2=R*SIN(W)
590   T2=S*COS(Q)
600   Z2=S*SIN(Q)
610 X3=(X2*V+64):Y3=(Y2+24):T3=(T2*V+64):Z3=(Z2+24):GOS
     UB 650
620 NEXT J
630 RETURN
640 REM LEVEL II LINE PLOT SUBROUTINE
650 IF X3=T3 THEN 810
660 IF X3>T3 THEN M=(Y3-Z3)/(X3-T3) ELSE M=(Z3-Y3)/(T3-
     X3)
670 B=Y3-M*X3
680 IF ABS(Z3-Y3) >ABS(T3-X3) THEN GOTO 750
690 IF X3>T3 THEN D=-1 ELSE D=1
700 FOR H=X3 TO T3 STEP D
710   Y3=M*H+B
720   SET (H,Y3)
730 NEXT H
740 RETURN
750 IF Y3>Z3 THEN D=-1 ELSE D=1
760 FOR H=Y3 TO Z3 STEP D
770   X3=(H-B)/M
780   SET(X3,H)
790 NEXT H
800 RETURN
810 IF Y3>Z3 THEN D=-1 ELSE D=1
820 FOR H=Y3 TO Z3 STEP D
830   SET(X3,H)
840 NEXT H
850 RETURN
```

Program Listing

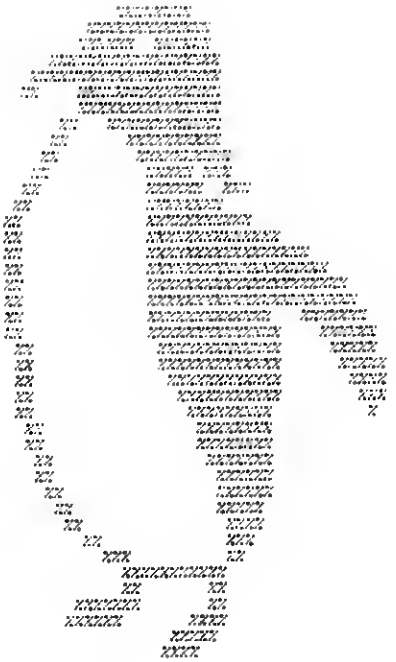
```
*** SNOWFLAKES ***
for TRS-80 Model I 16K
with MICROSOFT LEVEL III BASIC
10 CLS:PRINT"THIS PROGRAM RUNS UNDER LEVEL III BASIC"
20 PRINT"(MICROSOFT CONSUMER PRODUCTS, BELLVUE, WA.)"
30 PRINT"IF YOU HAVE NOT LOADED LEVEL III,"
40 PRINT"DO SO NOW, AND THEN RE-LOAD THIS PROGRAM."
```

```
50 PRINT"TO RUN THE PROGRAM, PRESS ENTER."
60 INPUT Q$
70 CLS:PRINT:PRINT
80 PRINT"SNOWFLAKES"
90 PRINT"COPYRIGHT 1980 BY"
100 PRINT"VALERIE VANN"
110 PRINT"631 G ST., DAVIS, CA."
120 FOR J=1 TO 800:NEXT J
130 REM TITLE PAGE
140 CLS
150 PRINT CHR$(23)
160 FOR J=2 TO 442 STEP 8
170  PRINT@J,"*"
180 FOR F=1TO10:NEXT F
190 NEXT J
200 PRINT@450,"* *  S N O W F L A K E S  * *"
210 FOR J=514 TO 958 STEP 8
220  PRINT@J,"*"
230  FOR F=1 TO 10 :NEXT F
240 NEXT J
250 FOR J=1 TO 950:NEXT J
260 REM INITIATE & SET NO. OF REPEATS
270 RANDOM:U=120/48
280 P=3.141592654
290 FOR E=1 TO 5
300 CLS
310 REM DEFINE LINE & NO. OF LINES
320 FOR K=1 TO (RND(5)+7)
330   X=RND(24)
340   Y=RND(24)
350   R=SQR(X[2+Y[2)
360   IF R>24 THEN 330
370   T=RND(24)
380   Z=RND(24)
390  S=SQR(T[2+Z[2)
400  IF S>24 THEN 370
410  GOSUB 520
420 REM MIRROR IMAGE OF LINE
430  Y=-1*Y
440  Z=-1*Z
450  GOSUB 520
460 FOR L=1 TO 80:NEXT L
470 NEXT K
480 FOR I=1 TO 2000:NEXT I
490 NEXT E
500 GOTO 140
510 REM SUBROUTINE-PLOT & ROTATE
520 W=ATN(Y/X)
530 Q=ATN(Z/T)
540 FOR J=1 TO 6
550   W=W+P/3
560   Q=Q+P/3
570   X2=R*COS(W)
580   Y2=R*SIN(W)
590   T2=S*COS(Q)
600   Z2=S*SIN(Q)
610   LINE(X2*U+64,Y2+24)-(T2*U+64,Z2+24),SET
620 NEXT J
630 RETURN
```



The gift that keeps on giving all year long.

CAL81



John F. Strazzarino
150 Dundee Drive
South San Francisco, CA 94080

Editor's Note: Here is a complete program for a 1981 calendar called CAL81. It includes a personalized option as well as a picture option. The first one lets you select from five pictured calendar heads: Mickey Mouse, butterfly, seal with a ball on its nose, airplane or penguin.

The second option lets you put in a personalized phrase at the bottom. The program has five phrases from which you can choose—or allows you to write your own.

When the author shows off his TRS-80 to friends, he likes to hand out a calendar with a personal message.

So, here is a program listing with samples of calendar graphics to go with the months. Merry Christmas!■

```
   1  1030 1130 1220 1320 1330 1340 1450 1460 1510 2690
   2  1350 2650
   3  1330 1360
   4  1370
   5  1130 1380
   6  1320 1390
  32  1530
  78  1400
  80  1140 2650
  99  1500
 500  1000
1000  2690
1010  1020
1120  1130
1160  1180
1190  1170
1200  1030 1210 1400 1450
1310  1320
1330  1330
1430  1440
1460  1220
1490  1510 1540
2310  1460 1500
2670  2680
2710  2380 2440 2510 2580
   A  1200/$ 1210/$ 1220/$ 2670/$ 2680/$ 2690/$
   B  1330/$2 1340/$ 1360/$
   C  1010/$ 1020/$ 1030/$ 1460/$
   O  1160 1170 1220/$ 1340/$ 1350/$ 1360/$ 1370/$ 1380/$ 1390/$ 1400/$ 1420/$
      2650/$ 2680/$
   E  1430/$ 1440/$ 1450/$ 1470 1510
   F  1120 1130/2 1140 1150 1190 1490 1500 1510 1530
   G  1310 1320/2 1330/2 1340 1350 1360 1370 1380 1390 1520 1530
   H  1470 1530
   I  2650 2660
   T  1140 1170
```

Program Listing 2. CAL81 Cross references

```
1    9999    8888    1
1    9  9    8  8    1
1    9999    8888    1
1       9    8  8    1
1    9999    8888    1
```

FEBRUARY

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

MARCH

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

APRIL

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

MF

S	M	T	W	T	F	S
3	4	5	6			
10	11	12	13			
17	18	19	20			
24	25	26	27			
31						

AUG

S	M	T	W	T	F	S
			1	2		
5	6	7	8	9		
12	13	14	15	16		
19	20	21	22	23		
26	27	28	29	30		

M

S	M	T	W	T	F	S
2	3	4				
9	10	11				
16	17	18				
23	24	25				
30						

OCTOBER

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

NOV

S	M	T	W	T	F	S
1	2	3				
8	9	10	1			
15	16	17	1			
22	23	24	2			
29	30					

Program Listing 1

```
1000 RESTORE:CLEAR 500:CLS:PRINT"WELCOME TO THE CALENDA
     R MAKER":PRINT
1010 INPUT "DO YOU WANT A PICTURE ON YOUR CALENDAR";C$
1020 IF C$="" THEN 1010
1030 IF LEFT$(C$,1)<>"Y" THEN 1200
1040 PRINT:PRINT"WE HAVE FIVE PICTURES FROM WHICH TO CH
     OOSE:"
1050 PRINT
1060 PRINT"1) MICKEY MOUSE"
1070 PRINT"2) BUTTERFLY"
1080 PRINT"3) SEAL AND BALL"
1090 PRINT"4) AIRPLANE"
1100 PRINT"5) PENGUIN"
1110 PRINT
1120 INPUT "WHICH NUMBER PICTURE DO YOU WANT";F
1130 IF F<1 OR F>5 THEN PRINT "MUST BE FROM 1 TO 5":GOT
     O 1120
1140 T=80+F
1150 PRINT:PRINT "LOOKING FOR PICTURE #";F
1160 READ D
1170 IF D=T THEN 1190
1180 GOTO 1160
1190 PRINT"PICTURE #";F;" FOUND"
1200 PRINT:INPUT"DO YOU WANT PERSONALIZATION";A$
1210 IF A$=""THEN 1200
1220 IF LEFT$(A$,1)<>"Y" THEN D$="":GOTO 1460
1230 PRINT:PRINT"THE FOLLOWING PERSONALIZATIONS ARE AVA
     ILABLE:"
1240 PRINT
1250 PRINT"1) THIS CALENDAR MADE ESPECIALLY FOR (YOUR N
     AME)"
1260 PRINT"2) THIS CALENDAR WAS MADE BY A HOME COMPUTER
     "
1270 PRINT"3) PRODUCED FOR (YOUR NAME) BY A TRS-80 MODE
     L I"
1280 PRINT"4) MERRY CHRISTMAS ** HAPPY NEW YEAR"
1290 PRINT"5) BEST WISHES FOR A HAPPY AND PROSPEROUS NE
     W YEAR"
1300 PRINT"6) MAKE YOUR OWN PHRASE"
1310 PRINT:INPUT "WHICH PERSONALIZATION NUMBER";G
1320 IF G<1 OR G>6 PRINT "MUST BE BETWEEN 1 AND 6":GOTO
     1310
1330 IF G=1 OR G=3 LINEINPUT "NAME? ";B$:IF B$=""THEN 1
     330
1340 IF G=1 THEN D$="THIS CALENDAR MADE ESPECIALLY FOR
     "+B$
1350 IF G=2 THEN D$="THIS CALENDAR WAS MADE BY A HOME C
     OMPUTER"
1360 IF G=3 THEN D$="PRODUCED FOR "+B$+" BY A TRS-80 MO
     DEL I"
1370 IF G=4 THEN D$="MERRY CHRISTMAS ** HAPPY NEW YEAR"
1380 IF G=5 THEN D$="BEST WISHES FOR A HAPPY AND PROSPE
     ROUS NEW YEAR"
1390 IF G=6 THEN LINE INPUT "WHAT IS YOUR PHRASE? ";D$
1400 IF LEN(D$)>70 THEN PRINT "LINE TOO LONG":GOTO 1200
1410 PRINT:PRINT"YOUR PERSONALIZATION WILL BE AS FOLLOW
     S:"
1420 PRINT : PRINT D$ :PRINT
```

Program continues

```
1430 INPUT "IS THIS OK";E$
1440 IF E$="" THEN 1430
1450 IF LEFT$(E$,1)<>"Y" THEN 1200
1460 IF LEFT$(C$,1)<>"Y" THEN 2310
1470 READ E,H
1480 LPRINT" ":LPRINT" ":LPRINT" "
1490 READ F
1500 IF F=99 THEN 2310
1510 IF F= -1 THEN LPRINT" ":LPRINT TAB(E);:GOTO 1490
1520 READ G
1530 LPRINT STRING$(F,32);STRING$(G,H);
1540 GOTO 1490
1550 'MOUSE DATA
1560 DATA 81,10,80
1570 DATA -1,43,6,-1,42,8,-1,41,10,-1,41,10,-1,3,4,34,1
     0,-1,1,10,4,4,23,8,-1
1580 DATA 0,13,1,3,26,6,-1,0,16,17,15,-1,1,15,14,18,-1,
     2,14,7,2,3,2,2,4,2,10,-1
1590 DATA 4,14,4,3,2,1,4,2,4,11,-1,8,11,3,6,2,4,2,14,3,
     6,-1
1600 DATA 13,7,2,29,1,8,-1,18,42,-1,20,40,-1,21,39,-1,2
     4,13,3,11,1,7,-1
1610 DATA 25,6,6,1,2,7,6,5,-1,20,4,2,4,2,4,-1,30,3,5,14
     ,-1,31,22,-1
1620 DATA 29,12,10,4,-1,27,13,11,4,-1,25,15,11,4,-1
1630 DATA 24,18,7,4,-1,23,20,4,7,-1,22,2,2,29,1,2,-1
1640 DATA 21,2,3,31,-1,21,3,2,4,2,23,-1,20,9,3,13,1,8,-
     1,20,10,2,12,2,2,4,4,-1
1650 DATA 6,4,10,24,2,3,4,2,-1,4,8,6,23,-1,3,10,4,5,14,
     4,-1,3,10,3,4,18,2,7,2,-1
1660 DATA 3,15,21,12,-1,4,13,24,10,-1,5,13,25,9,-1,7,11
     ,25,10,-1,8,9,24,12,-1
1670 DATA 9,7,23,13,-1,10,5,23,13,-1,99
1680 'BUTTERFLY DATA
1690 DATA 82,16,37
1700 DATA -1,20,1,7,1,-1,21,1,5,1,-1,21,1,5,1,-1
1710 DATA 21,1,5,1,-1,22,1,3,1,-1,22,1,3,1,-1
1720 DATA 22,1,3,1,-1,22,1,3,1,-1
1730 DATA 0,10,13,1,1,1,13,10,-1,8,3,6,4,10,1,1,1,10,4,
     6,3,-1
1740 DATA 0,2,1,6,1,5,8,1,1,1,8,5,1,6,1,2,-1
1750 DATA 0,2,6,3,2,5,5,1,1,1,5,5,2,3,6,2,-1
1760 DATA 1,2,1,3,2,3,1,5,1,3,1,1,1,1,3,1,5,1,3,2,3,1
     ,2,-1
1770 DATA 1,2,1,4,2,3,1,5,1,9,1,5,1,3,2,4,1,2,-1
1780 DATA 1,2,1,4,2,3,1,6,1,7,1,6,1,3,2,4,1,2,-1
1790 DATA 2,2,4,4,2,6,1,7,1,6,2,4,4,2,-1
1800 DATA 2,3,1,4,4,6,1,7,1,6,4,4,1,3,-1
1810 DATA 3,12,6,7,6,12,-1,4,6,4,6,2,5,2,6,4,6,-1
1820 DATA 5,3,3,6,3,1,1,5,1,1,3,6,3,3,-1,9,5,3,3,2,5,2,
     3,3,5,-1
1830 DATA 7,5,2,5,1,1,1,5,1,1,1,5,2,5,-1
1840 DATA 7,4,1,6,1,2,1,5,1,2,1,6,1,4,-1
1850 DATA 8,3,1,5,1,2,3,3,3,2,1,5,1,3,-1
1860 DATA 9,1,1,5,1,3,3,3,3,3,1,5,1,1,-1
1870 DATA 10,5,1,4,3,3,3,4,1,5,-1,11,4,1,4,3,3,3,4,1,4,
     -1
1880 DATA 12,2,1,4,4,3,4,4,1,2,-1,15,4,4,3,4,4,-1
1890 DATA 17,2,5,1,5,2,-1,99
1900 'SEAL AND BALL DATA
1910 DATA 83,16,37
1920 DATA -1,11,10,-1,8,16,-1,5,22,-1,3,26,-1,2,28,-1
1930 DATA 1,30,-1,0,32,-1,0,32,-1,0,32,-1,0,32,-1,0,32,
     -1
1940 DATA 0,32,-1,1,30,-1,2,28,-1,3,26,-1,5,22,-1,8,16,
     -1
1950 DATA 11,10,-1,15,1,-1,14,3,-1,14,4,-1,13,7,-1,13,3
     ,1,4,-1
1960 DATA 13,9,-1,13,10,-1,13,10,-1,12,10,-1,11,10,-1,1
     0,12,-1
1970 DATA 9,14,-1,9,17,-1,8,21,-1,8,24,-1,8,26,-1,8,28,
     -1
1980 DATA 8,30,-1,8,31,-1,8,32,-1,9,32,-1,9,32,-1,10,32
     ,-1
1990 DATA 10,32,-1,8,8,1,25,-1,6,8,5,23,-1,4,7,11,20,2,
     3,-1
2000 DATA 4,5,22,11,1,4,-1,3,3,26,6,1,8,-1
2010 DATA 32,6,2,5,-1,31,6,-1,30,5,-1,29,5,-1,99
2020 'AIRPLANE DATA
2030 DATA 84,10,37
2040 DATA -1,45,5,-1,43,8,-1,41,10,-1,40,3,3,5,-1,39,3,
     1,3
2050 DATA 1,4,-1,39,2,1,5,1,3,-1,38,3,1,5,1,2,-1,37,5,1
     ,3,1,3,-1
2060 DATA 36,7,3,3,-1,35,13,-1,35,13,-1,26,5,3,13,-1,26
     ,6,1,14,-1
2070 DATA 27,19,-1,27,19,-1,28,17,-1,29,16,-1,22,5,2,15
     ,-1,22,6,1,15,-1
2080 DATA 13,4,6,20,-1,11,8,4,19,-1,10,10,4,18,-1,10,12
     ,3,16,-1,10,14,1,15,-1
2090 DATA 10,29,-1,11,27,-1,13,25,-1,14,23,-1,16,21,-1,
     18,18,-1,11,4,3,20,-1
2100 DATA 10,7,1,22,-1,10,31,-1,11,31,-1,12,31,-1,13,32
     ,-1,6,4,3,33,-1
2110 DATA 6,6,1,15,2,18,-1,6,21,6,16,17,2,-1,7,19,10,15
     ,14,3,-1
2120 DATA 8,17,14,13,11,5,-1,8,16,17,13,6,7,-1,7,16,20,
     13,1,9,-1,6,16,23,20,-1
2130 DATA 5,16,26,17,-1,4,16,29,14,-1,3,16,32,11,-1,3,1
     5,34,9,-1,2,15,34,11,-1
```

Program continues

```
2140 DATA 1,3,3,9,35,11,-1,1,2,1,3,1,7,36,11,-1,0,2,1,5
     ,1,5,30,6,2,2,-1
2150 DATA 0,2,1,5,1,4,38,7,-1,0,3,1,3,1,4,39,6,-1,0,4,3
     ,3,41,5,-1,0,9,42,4,-1
2160 DATA 1,7,43,3,-1,2,5,44,2,-1,51,1,99
2170 'PENGUIN DATA
2180 DATA  85,25,37
2190 DATA -1,12,8,-1,9,13,-1,8,6,2,6,-1
2200 DATA 5,18,-1,3,20,-1,2,2,4,15,-1,8,15,-1,6,2,3,12,
     -1
2210 DATA 5,2,6,10,-1,4,2,8,10,-1,3,2,10,5,1,3,-1
2220 DATA 2,2,11,6,2,3,-1,1,2,12,8,-1,0,2,13,11,-1
2230 DATA 0,2,13,14,-1,0,2,13,17,-1,0,2,13,19,-1
2240 DATA 0,2,13,21,-1,0,2,13,22,-1,0,2,13,13,3,7,-1
2250 DATA 0,2,13,14,4,6,-1,1,2,13,13,5,5,-1,1,2,13,13,6
     ,5,-1
2260 DATA 1,2,14,12,7,4,-1,1,2,15,11,8,3,-1,1,2,16,9,10
     ,1,-1
2270 DATA 2,2,16,8,-1,2,2,16,8,-1,3,2,16,7,-1,3,2,17,6,
     -1
2280 DATA 4,2,16,6,-1,5,2,15,5,-1,6,2,15,4,-1,0,2,13,3,
     -1
2290 DATA 10,3,10,2,-1,12,11,-1,12,2,7,2,-1,7,7,7,2,-1
2300 DATA 6,6,7,4,-1,17,5,-1,16,4,-1,99
2310 LPRINT " ":LPRINT " ":LPRINT" "
2320 LPRINT"                                   1    9999    88
     00     1"
2330 LPRINT"                                   1    9  9    8
       8    1"
2340 LPRINT"                                   1    9999    00
     88     1"
2350 LPRINT"                                   1       9    8
       0    1"
2360 LPRINT"                                   1    9999    88
     88     1"
2370 LPRINT" ":LPRINT" ":LPRINT" "
2380 LPRINT"        JANUARY                                FEBRUAR
     Y                      MARCH":GOSUB 2710
2390 LPRINT"                1  2  3          1  2  3  4  5
       6  7             1  2  3  4  5  6  7"
2400 LPRINT" 4  5  6  7  8  9 10          0  9 10 11 12
      13 14          8  9 10 11 12 13 14"
2410 LPRINT"11 12 13 14 15 16 17         15 16 17 18 19
      20 21         15 16 17 18 19 20 21"
2420 LPRINT"18 19 20 21 22 23 24         22 23 24 25 26
      27 28         22 23 24 25 26 27 28"
2430 LPRINT"25 26 27 28 29 30 31
                    29 30 31":LPRINT" "
2440 LPRINT"         APRIL                                  MAY
                          JUNE":GOSUB 2710
2450 LPRINT"           1  2  3  4
         1  2           1  2  3  4  5  6"
2460 LPRINT" 5  6  7  8  9 10 11          3  4  5  6  7
       8  9          7  8  9 10 11 12 13"
2470 LPRINT"12 13 14 15 16 17 18         10 11 12 13 14
      15 16         14 15 16 17 18 19 20"
2480 LPRINT"19 20 21 22 23 24 25         17 18 19 20 21
      22 23         21 22 23 24 25 26 27"
2490 LPRINT"26 27 28 29 30               24 25 26 27 28
      29 30         28 29 30"
2500 LPRINT"                                  31":LPRINT" "
2510 LPRINT"        JULY                                   AUGUST
                            SEPTEMBER":GOSUB2710
2520 LPRINT"             1  2  3  4
           1                    1  2  3  4  5"
2530 LPRINT" 5  6  7  8  9 10 11          2  3  4  5  6
       7  8          6  7  8  9 10 11 12"
2540 LPRINT"12 13 14 15 16 17 18          9 10 11 12 13
      14 15         13 14 15 16 17 18 19"
2550 LPRINT"19 20 21 22 23 24 25         16 17 18 19 20
      21 22         20 21 22 23 24 25 26"
2560 LPRINT"26 27 28 29 30 31            23 24 25 26 27
      28 29         27 28 29 30"
2570 LPRINT"                                  30 31":LPRINT"
     "
2580 LPRINT"        OCTOBER                                NOVEMBE
     R                    DECEMBER"
2590 GOSUB 2710
2600 LPRINT"                1  2  3          1  2  3  4  5
       6  7                1  2  3  4  5"
2610 LPRINT" 4  5  6  7  8  9 10          8  9 10 11 12
      13 14          6  7  8  9 10 11 12"
2620 LPRINT"11 12 13 14 15 16 17         15 16 17 18 19
      20 21         13 14 15 16 17 18 19"
2630 LPRINT"18 19 20 21 22 23 24         22 23 24 25 26
      27 28         20 21 22 23 24 25 26"
2640 LPRINT"25 26 27 28 29 30 31         29 30
                    27 28 29 30 31"
2650 I=(80-LEN(D$))/2
2660 LPRINT" ":LPRINT" ":LPRINTTAB(I)D$:LPRINT" ":LPRIN
     T" "
2670 PRINT:INPUT "DO YOU WANT TO MAKE ANOTHER CALENDAR"
     ;A$
2680 IF A$="" THEN 2670
2690 IF LEFT$(A$,1)="Y" THEN 1000
2700 PRINT:PRINT"END OF PROGRAM":PRINT:END
2710 LPRINT" "
2720 LPRINT" S  M  T  W  T  F  S          S  M  T  W  T
       F  S          S  M  T  W  T  F  S"
2730 LPRINT" -  -  -  -  -  -  -          -  -  -  -  -
       -  -          -  -  -  -  -  -  -"
2740 RETURN
```

Be different!
Send an electronic Christmas card with an 80 hallmark this year.

Holiday Cheer

Norman S. Kerr
1571 Burton St.
St. Paul, MN 55108

Last Christmas, I decided to let Max, my friendly TRS-80, write personalized holiday letters. You can do the same! The two programs you need are a computerized address list and a letter writing program.

I have written an address list program that stores information in a two-dimensional array (Table 1). I have used a salutation entry permitting you to use "Norman S." in the mailing address while using "Sylvie and Norman" in the salutation of your yuletide letter. "Special Intereste" allows you to include a statement about each person's hobby or profession. (If you can't think of anything to say here, the letter writing program will substitute "leisure time" whenever this statement is missing).

The subroutine in lines 6140-6190 permits you to remove names from your address list.

Program Listing 1 produces a copy of the address list on your line printer. It also allows you to record your receipt of cards at the end of the season.

Updating

Once you have your Christ-

VALUE OF J	LISTED PROGRAM
1	LAST NAME
2	FIRST NAME
3	STREET ADDRESS
4	CITY
5	STATE
6	ZIP CODE
7	CARD RECEIVED?
8	SALUTATION
9	SPECIAL INTEREST

Table 1

```
                    CHRISTMAS 1980


DEAR UNCLE WAYNE,

     MERRY CHRISTMAS.  THE KERRS ASKED ME TO WRITE THEIR
WINTER SOLSTICE LETTER FOR THEM THIS YEAR.
     I HOPE YOU HAVE HAD AS GOOD A YEAR AT 80 PINE STREET
AS WE HAVE HAD AT 1571 BURTON STREET.  I DO NOT WISH ON YOU
IN PETERBOROUGH, NEW HAMPSHIRE AS MUCH COLD WEATHER AS WE
HAVE HAD.

     NOTE THAT I HAVE LEFT SPACES TO INDICATE A NEW PARAGRAPH.
IF YOU WRITE LINES THAT ARE APPROXIMATELY THE WIDTH TO BE
PRINTED YOU WILL SAVE A GOOD DEAL OF TIME PRINTING OUT YOUR
 CHRISTMAS LETTER, AS THE COMPUTER WILL NOT HAVE TO
EXTENSIVELY PROCESS EACH LINE BEFORE IT SENDS IT TO THE
LINE PRINTER.

     BE CERTAIN TO MENTION NEWS ABOUT EACH MEMBER OF THE FAMILY.
     NORMAN HAS HAD AN ARTICLE PUBLISHED IN 80-MICROCOMPUTING.
     WE HOPE YOU HAVE BEEN ENJOYING YOUR PUBLISHING
MAGAZINES DURING THE PAST YEAR.
     HERE'S HOPING YOU HAVE HAPPY HOLIDAYS AT PETERBOROUGH
AND A HAPPY AND PROSPEROUS NEW YEAR.

                         MAX
```

Program Listing 1

```
1 REM CHRISTMAS ADDRESS PROGRAM
2 REM BY NORMAN S. KERR
3 REM 1571 BURTON STREET
4 REM ST. PAUL, MINNESOTA 55108
5 REM TO BE USED TOGETHER WITH CHRISTMAS LETTER PROGRAM
```

Program continues

✓ Reader Service—see page 274

```
10 CLEAR 1000
20 DIM A$(100,9), Z$(9)
30 Z$(1)="LAST NAME": Z$(2)= "FIRST NAME"
40 Z$(3)="STREET ADDRESS": Z$(4)="CITY": Z$(5)= "STATE"
50 Z$(6)= "ZIP CODE": Z$(7)= "CARD RECEIVED"
60 Z$(8)= "SALUTATION": Z$(9)="SPECIAL INTERESTS"
100 CLS
110 PRINT:PRINT"MENU - "
120 PRINT,,"1 RETRIEVE LIST FROM CASSETTE"
130 PRINT,,"2 SAVE LIST ON CASSETTE"
140 PRINT,,"3 DISPLAY LIST"
150 PRINT,,"4 ADD NAMES TO LIST"
160 PRINT,,"5 EDIT LIST"
170 PRINT,,"6 ALPHABETIZE LIST"
175 PRINT,,"7 FINISHED WITH PROGRAM"
180 INPUT "MAKE YOUR CHOICE";A
190 ON A GOTO 1000,2000,3000,4000,5000,6000,7000
1000 REM RETRIEVE LIST FROM CASSETTE
1010 CLS: PRINT@ 320, "LOAD TAPE AND PRESS 'PLAY'"
1020 PRINT:INPUT "HIT 'ENTER' TO CONTINUE";D
1030 IN=0
1040 I=IN:PRINT:PRINT"TAPE READ IN PROGRESS."
1050 I=I+1: INPUT@-1,A$(I,1), A$(I,2),A$(I,3),A$(I,4),A
     $(I,5),A$(I,6),A$(I,7),A$(I,8),A$(I,9)
1060 IF A$(I,1)<>"END OF FILE" THEN  GOTO 1050
1070 I=I-1
1080 PRINT:PRINT I " RECORDS READ"
1090 INPUT "HIT 'ENTER' TO CONTINUE";D
1100 GOTO 100
2000 REM STORE LIST ON CASSETTE
2010 CLS:PRINT@320, "LOAD TAPE - PRESS 'PLAY' AND 'RECO
     RD'"
2020 INPUT "HIT 'ENTER' WHEN READY TO CONTINUE";D
2030 II=I
2040 A$(II+1,1)="END OF FILE"
2050 FOR I=1 TO II+1
2060 PRINT@-1,A$(I,1),A$(I,2),A$(I,3),A$(I,4),A$(I,5),A
     $(I,6),A$(I,7),A$(I,8),A$(I,9)
2070 NEXT I
2080 I=II
2090 PRINT:PRINT I "RECORDS SAVED ON TAPE"
2100 INPUT "PRESS 'ENTER' TO CONTINUE";D
2110 GOTO 100
3000 REM DISPLAY LIST
3010 PRINT:PRINT "-1 DISPLAY LIST ON CRT"
3020 PRINT"-2 PRINT LIST ON LINE PRINTER"
3030 PRINT"-3 RETURN TO MAIN MENU"
3040 INPUT "MAKE YOUR CHOICE (1, 2, OR 3)";A
3050 ON A GOTO 3100,3200,100
3100 REM DISPLAY LIST ON CRT
3120 FOR I=1 TO II
3130 FOR J=1 TO 9
3140 PRINT Z$(J)":";
3150 PRINT,A$(I,J)
3160 NEXT J
3170 INPUT "PRESS 'ENTER' TO CONTINUE";D
3180 CLS: NEXT I
3190 I=II:GOTO 100
3200 REM PRINT ON LINE PRINTER
3210 II=I
3220 CLS:PRINT"MAKE CERTAIN LINE PRINTER IS TURNED ON."
3230 INPUT "PRESS 'ENTER' TO CONTINUE";D
3240 FOR J=1 TO 9
3250 LPRINT Z$(J)
3260 NEXT J
3270 LPRINT
3280 FOR I=1 TO II
3290  FOR J=1 TO 9
3300  LPRINT A$(I,J)
3310  NEXT J
3320  LPRINT
3330 NEXT I
3340 I=II
3350 LPRINT CHR$(11)
3360 GOTO 3000
4000 REM ADD NAMES TO LIST
4010 CLS: I=I+1
4020 FOR J=1 TO 9
4030 PRINT Z$(J);
4040 INPUT A$(I,J)
4050 NEXT J
4060 PRINT:PRINT,"-1 SAVE THIS ENTRY"
4070 PRINT,"-2 EDIT THIS ENTRY"
4080 PRINT,"-3 DISCARD THIS ENTRY"
4090 INPUT "CHOOSE 1, 2, OR 3";A
4100 ON A GOTO 4200,4300,4400
4200 REM SAVE THIS ENTRY
4210 CLS: PRINT:PRINT,"-1 MAKE AN ADDITIONAL ENTRY"
4220 PRINT,"-2 RETURN TO MAIN MENU"
4230 INPUT "CHOOSE 1 OR 2";A
4240 ON A GOTO 4000,100
4300 REM EDIT THIS ENTRY
4310 GOTO 5400
4400 REM DISCARD THIS ENTRY
4410 I=I-1: GOTO 4200
5000 REM EDIT ENTRIES
5010 PRINT:PRINT,"-1 EDIT ENTIRE LIST"
5020 PRINT,"-2 EDIT ONE ITEM
5030 INPUT "CHOOSE 1 OR 2";A
5040 ON A GOTO 5100,5300
5100 REM EDIT ENTIRE LIST
5110 CLS: II=I
5120 FOR I=1 TO II
5130 FOR J=1 TO 9
5140 PRINT J; +" "+Z$(J)+": "+A$(I,J)
5150 NEXT J
5160 PRINT"10 FINISHED EDITING"
5165 PRINT" 11 EDIT NEXT ITEM"
5170 INPUT "WHICH ITEM DO YOU WISH TO CHANGE";A
5180 IF A=10 GOTO 100
5185 IF A=11 THEN NEXT I
5190 INPUT A$(I,A)
5200 GOTO 5130
5300 REM EDIT ONE ITEM
5310 CLS: INPUT "LAST NAME OF ITEM TO BE EDITED"; B$
5320 II=I
5330 FOR I=1 TO II
5340 IF A$(I,1)=B$ THEN GOTO 5400
5350 NEXT I
5360 I=II
5370 PRINT"SORRY THAT ITEN NOT FOUND"
5380 INPUT "PRESS 'ENTER' TO CONTINUE";D
5390 GOTO 100
5400 REN EDIT ONE ITEN
5410 FOR J=1 TO 9
5420 PRINT J" "+Z$(J)+": "+A$(I,J)
5430 NEXT J
5440 PRINT "10 - FINISHED EDITING"
5450 INPUT "MAKE YOUR CHOICE";A
5460 IF A=10 THEN GOTO 100
5470 PRINT "REENTER LINE "A
5480 INPUT A$(I,A)
5490 GOTO 5410
6000 REN ALPHABETIZE LIST
6010 CLS:PRINT:PRINT"ALPBABETIZING
6020 IH=I
6030 FOR I=1 TO IH-1
6040 FOR J=I TO 9: A$(0,J)=A$(I,J):NEXT J:IL=I
6050 FOR II=I+1 TO IN
6060 IF A$(0,1)<=A$(II,1) THEN GOTO 6080
6070 FOR J=1 TO 9: A$(0,J)=A$(II,J):NEXT J: IL=II
6080 NEXT II
6090 FOR J=1 TO 9: A$(0,J)=A$(IL,J): NEXT J
6100 FOR J=1 TO 9: A$(IL,J)=A$(I,J):NEXT J
6110 FOR J=1 TO 9: A$(I,J)=A$(0,J): NEXT J
6120 NEXT I
6125 I=IH
6130 II=I: I=1
6140 IF A$(I,1)<>"" THEN I=II: GOTO 100
6150 FOR I=1 TO II
6160 FOR J=1 TO 9: A$(I,J)=A$((I+1),J): NEXT J
6170 NEXT I
6180 I=II-1
6190 GOTO 6130
7000 REM EXIT PROGRAM
7010 PRINT"HAVE YOU SAVED YOUR LIST ON CASSETTE"
7020 PRINT"-1 YES        -2 NO"
7030 INPUT "ENTER 1 OR 2";A
7040 IF A<>1 THEN GOTO 100
7050 PRINT "GOODBYE FOR NOW.  hOPE I CAN NORK NITH YOU
     LATER".
```

Program Listing 2

```
1 REM CHRISTMAS LETTER PROGRAM
2 REM BY NORMAN S. KERR
3 REM 1571 BURTON STREET
4 REM ST. PAUL, MINNESOTA 55108
5 REM TO BE USED TOGETHER WITH CHRISTMAS ADDRESS LIST
10 CLEAR 10000: DEFINT A-Z: DIM A$(100,9), B$(50)
20 CLS: PRINT@ 320, "LOAD CHRISTMAS LIST ADDRESS TAPE A
   ND PRESS 'PLAY'"
30 INPUT "HIT 'ENTER' TO CONTINUE"; D
40 IN=0
50 I=IN: PRINT: PRINT"TAPE READ IN PROGRESS."
60 I=I+1: INPUT@-1,A$(I,1), A$(I,2), A$(I,3), A$(I,4),
   A$(I,5), A$(I,6), A$(I,7), A$(I,8), A$(I,9)
70 IF A$(I,1)<> "END OF FILE" THEN 60
75 I=I-1
80 PRINT:PRINT I "RECORDS READ"
90 INPUT "HIT 'ENTER' TO CONTINUE";D
100 CLS: GOTO 10000
1000 IF A$(I,9)=" " THEN A$(I,9)="LEISURE TIME"
1010 B$(0) = "DEAR " + A$(I,8) +","
1020 B$(1) = " "
1030 B$(2)="     MERRY CHRISTMAS.  THE KERRS ASKED ME T
     O WRITE THEIR"
1040 B$(3)="WINTER SOLSTICE LETTER FOR THEN THIS YEAR."
1050 B$(4)="     I HOPE YOU HAVE HAD AS GOOD A YEAR AT
     "+A$(I,3)
1060 B$(5)="AS WE HAVE HAD AT 1571 BURTON STREET.  I DO
```

Program continues

I'M A BELIEVER !!

"I Love it !!. . .It's really a incredible O/S. It' just great! Now I see why people who have seen it say they are now believers. I know I am." *LANCE MICKLUS*

1) Large (8") drive support.
2) Double Sided drive support.
3) Double Density drive support.
4) 80 Track drive support.
*NOTE all above drives may be mixed on any one system and can be configured at Sysgen time or during any Backup!
5) Winchester technology fixed drive support.
6) Supports any combination of the above drives up to a max. of 8 drives.
7) Supports double-speed processor clock modifications. (Archbold for example)
8) FASTER! —— Improved overlay structure using ISAM accessing techniques improves loading times by up to 1400%.
9) General purpose output spoolers of a true, symbiont design provide simultaneous output and program execution without any user intervention.
10) Keyboard Type-Ahead feature permits you to enter keystrokes before your programs need them.
11) User definable keys, all 26 letters.
12) Built in Graphic string packer lets you enter graphic symbols into a BASIC program from the keyboard through the use of the (Clear) key. The (Clear) key is simply held down (just like the (Shift) keys) during other keystrokes and viola...graphics!
13) Dated files. —— All files are accompanied by the date of their last modification (creation or write).
14) Marked files. —— All files are accompanied by a 'mark' if they have been modified since they were last backed up. This permits the BACKUP utility to copy only those files which have actually been updated since a previous backup.
15) File transfer by class. Allows transferring of all files of a similar directory classification such as /CMD, /BAS, /PCL, etc.

VTOS 4.0

VTOS 4.0
Operating System Diskette with Operator's Guide
$99.95

VTOS 4.0
Master Reference Manual
$29.95

VTOS 4.0
Combination - 4.0 disk, Operator's Guide, and Master Reference Manual
$125.00

16) Built-in SYSTEM command contains lower case display driver, screen print, break key disable, blinking cursor, disk drive stepping rate and motor-on delay modifications, and more.
17) User may SYSGEN a custom VTOS system configuration containing special I/O drivers, device LINKing and ROUTEing, SPOOLing and DEBUG tasks, etc. which will be automatically loaded during the BOOT process without requiring a more lengthy AUTO and CHAIN procedure.
18) Non-BREAKable AUTO and CHAIN commands.
19) Wild-card DIRectory. Permits you to locate all files of a certain classification such as '/BAS'. Uniformly indicates file size in K (1024 bytes) regardless of drive type. "DIR 0" would give you all your files that start with "0".
20) Dynamic file name defaults in APPEND, COPY, and RENAME commands allow you to specify only minimal information about file names.
21) COPY and APPEND commands execute up to 300% faster.
22) ALLOCate command for pre-allocation and non-releasibility of file space. File space will never shrink if this option used.
23) MEMORY command for directly setting upper memory limit.
24) Variable Length file support is incorporated which automatically blocks short user data records both within a sector and across sector boundaries thereby taking maximum advantage of disk file space.
25) No security disk needed to make backups or to run the system!
26) Though many O/S bear his design and code VTOS 4.0 is the only Fully Aproved Operating System by Randy Cook! And it is FANTASTIC!
27) Endorsed by Scott Adams and Lance Micklus!

Available from the following distributors or your local computer store. *DEALER INQUIRIES INVITED.*

5% Discount Just For Mentioning This Ad. (Valid month of this publication ONLY)

```
      NOT WISH ON YOU"
1070 B$(6)="IS "+A$(I,4)+", "+A$(I,5)+" AS MUCH COLD WE
     ATHER AS WE HAVE HAD."
1080 B$(8)="     NOTE THAT I HAVE LEFT SPACES TO INDICA
     TE A NEW PARAGRAPH."
1090 B$(9)="IF YOU WRITE LINES THAT ARE APPROXIMATELY T
     HE WIDTH TO BE"
1100 B$(10)="PRINTED YOU WILL SAVE A GOOD DEAL OF TIME
     PRINTING OUT YOUR"
1110 B$(11)=" CHRISTMAS LETTER, AS THE COMPUTER WILL NO
     T HAVE TO"
1120 B$(12)="EXTENSIVELY PROCESS EACH LINE BEFORE IT SE
     NDS IT TO THE LINE PRINTER."
1130 B$(13)=" "
1140 B$(14)="     BE CERTAIN TO MENTION NEWS ABOUT EACH
      MEMBER OF THE FAMILY:"
1150 B$(15)="     NORMAN HAS HAD AN ARTICLE PUBLISHED I
     N 8B-HICROCOMPUTING."
1160 B$(16)="     WE HOPE YOU HAVE BEEN ENJOYING YOUR "
     + A$(I,9)+" DURING THE PAST YEAR."
1170 B$(17)="    HERE'S HOPING YOU HAVE HAPPY HOLIDAYS
     AT "+A$(I,4)+" AND A HAPPY AND PROSPEROUS NEW YEAR
     ."
1180 B$(18)="  "
1190 B$(19)="                          MAX"
2000 RETURN
10000 REM PRINT ON LINE PRINTER
10B2B II=I
10830 FOR I=1 TO II
1084B GOSUB 10BBB
10850 LPRINT CHR$(14)"                CHRISTMAS 1980"
10860 LPRINT CHR$(15): LPRINT: LPRINT
10870 FOR N=0 TO 19: REM CHANGE TOP NUMBER TO NUMBER IN
     B$(X)
10080 LA=LEN(B$(N))
1009B IF LA<65 AA$=B$(N): GOTO 10250
10100 X=60
10110 AA$=LEFT$(B$(N),X)
10120 IF RIGHT$(AA$,1)<>CHR$(32) X=X-1: GOTO 10110
10130 LB=LA-LEN(AA$)
1014B IF LB<60 AB$=MID$(B$(N),(X+1),LB): GOTO 10250
10150 Y=60
10160 AB$=MID$(B$(N),(X+1),Y)
10170 IF RIGHT$(AB$,1)<>CHR$(32) Y=Y-1: GOTO 10160
10180 LC=LB-LEN(A0$)
10190 IF LC<60 AC$=MID$(B$(N),(X+Y+1),LC): GOTO 10250
10200 Z=60
10210 AC$=MID$(B$(N),(X+Y+1),Z)
10220 IF RIGHT$(AC$,1)<>CHR$(32) Z=Z-1: GOTO 10210
10230 LD=LC-LEN(AC$)
10240 AD$=MID$(B$(N),(X+Y+Z+1),LD)
10250 LPRINT"            "+ AA$
10260 IF LEN(AB$)>1 LPRINT "            "+AB$
10270 IF LEN(AC$)>1 LPRINT "            "+AC$
10280 IF LEN(AD$)>1 LPRINT"           "+ AD$

10290 AA$="": AB$="": AC$="": AD$=""
10300 NEXT N
10310 LPRINT CHR$(11)
1032B NEXT I
1033B I=II
1034B END
```

mas mailing list stored on a computer file, you need to update this file only once a year. As you receive cards from your friends, check their mailing addresses and make any necessary corrections. Then on a cold night in January transfer the necessary corrections to your computer file. If you wish, you can record the receipt of a card while in the 'edit entire file' mode.

Letter-writing (Program Listing 2) is quite simple. The address list is stored in the two-dimensional string array A$(I,J). The message to be printed is stored in the string array B$(W). Elements of the A$(I,J) array should be incorporated into B$(W) as often as possible—this is what personalizes your Christmas letters.

Once you have finished writing your letter, you will know the value of W, which must be changed in the line-print routine in line 10070. It is important that B$(W) arrays be reloaded after each letter has been printed and I has been incremented, so that the new elements from A$(I,J) will be printed with each letter.

Margin Routine

An important feature of the line printer routine, which may be useful in other programs, is included between lines 10080-10290. These lines prevent words from being split at the end of the printer's line. I arbitrarily set the margins at 10 spaces (lines 10260-10280) and set the printer's line at 60 characters and spaces in lines 10100, 10150, and 10200.

When typing in your B$(W) statements, you should try to cut down the use of this subroutine by keeping most lines at less than 60 characters. If you make too frequent use of this feature, you will think that your computer has crashed when the printer pauses (an understatement) in the middle of a letter. The program as presented in the Sample Letter causes no such hang-ups.

My TRS-80 is named Max. Substitute your computer's name here, or reword this portion of the letter to your taste. Max is a TRS-80, LEVEL II, with 32K RAM and an Anadex DP-8000 printer. The CHR$(14) in line 10050 causes the Anadex to print in boldface. The CHR$(15) in line 10060 restores it to normal printing.

Before attempting to produce individualized letters for your entire list, construct a dummy list containing, say, three entries. This will enable you to be certain your program is debugged, and allow you to set the top of page at the appropriate place on your printer.

I hope you will enjoy the program. If you don't send Christmas letters, send them out on Valentine's Day! ■

SAVE / add-ons for TRS-80® Software and Hardware

NEWDOS 80

A new enhanced NEWDOS for the TRS-80.

The most powerful Disk Operating System for the TRS-80, designed for the sophisticated user and professional programmer who demands the ultimate.

NEWDOS/80 is the planned upgrade from NEWDOS 2.1. Some of the features are:

- New BASIC commands for files with variable record lengths up to 4095
- Mix or match drives. Use 35, 40 or 80 track 5" disk drives or 8" disk drives, or combo.
- Security boot-up for BASIC or machine code application programs
- New editing commands
- Enhanced RENUMber that allows relocation
- Command chaining
- Device handling for routing to display and printer simultaneously
- DFG function, striking of D, F and G keys allows user to enter a mini-DOS without disturbing program
- Compatible with NEWDOS & TRSDOS
- Machine language Superzap/80 2.1 utilities and enhanced debug and copy

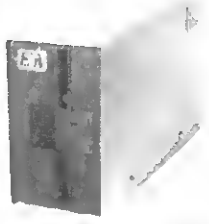
$149

NEW

TF-8 80 TRACK DISK DRIVE — Double Your Capacity — **$639**

TF-9 DUAL 80 TRACK DISK DRIVE — Quadruple Your Capacity — **$789**

FACTORY CLEARANCE



Demo single or dual head MPI disc drive, complete with Power Supply and Chassis. Full warranty

TF5D 40 TK **$279**

TDH-1D Dual 35 TK **$419**

LIMITED QUANTITIES

Announcing

8" Floppy Disk Drive System

for Model I

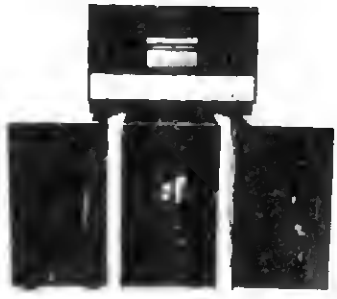
- One SA801 Floppy
- NEWDOS 80
- Cable & Adapter

$1095

Disk Drive Sale!

Complete with power supply and chassis

TF-3 Shugart SA400	$339
Pertec FD200, 40 track	$379
TF-5 MPI B51, 40 track	$369
TF-7 Micropolis 77 track	$574
TDH-1 Dual Sided drive 35 track	$499
TF-3M Drive Sys 2 Shugart	$698
NEWDOS+ 40 track **$110** 35 track	$99
Microconductor, Data Base Mgr Mod I **$249** Mod II	$399
AJA Business Pgk	$359
The Source	$100
Basic Compiler	$195



Disk Expansion System

• 2 Shugart SA400 TF-3	$718
• 1 Two-Drive Cable	$ 25
• 1 Expansion Interface 32K	$489
• 1 35-track DOS+	$ 99
TOTAL LIST PRICE	$1331
SPECIAL PRICE ONLY	**$1,149**
Same as above but includes TRS-80® Level II	**$1949**

MOD II 8" Disk System

• 1 Drive System	$949
• 2 Drive Expansion System	$1,445

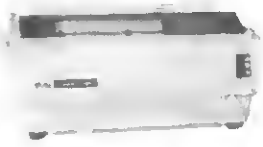
Drives for any Microcomputer

Does not include power supply & cabinet.

Pertec FD200	$282	FD250	$359
Shugart SA400	$279	SA800/801	$479
MPI B51	$279	B52	$349
MPI B91	$399	B92	$525

Printers

Centronic 779	$1,069
Base 2	$649
Centronics 737	$939
Centronics 702-9	$1,995
Anadex 9501	$1,549
Malibu	$2,495
Spinwriter	$2,549



Daisy Wheel **$1,779**

More Savings

INTRODUCTORY OFFER SAVE $300

TRS-80 Graphics List $949

OKIDATA MICROLINE 80 **$649**

Memory Kit (16K)	$49.00
AC Isolator (6 socket)	$49.95
Disk Head Cleaner	$19.95
Diskettes (10)	$30.00

Apparat, Inc.

4401 South Tamarac Parkway • Denver, CO 80237 • (303) 741-1779

MICROCOMPUTER TECHNOLOGY INCORPORATED

Order Desk Only 800-854-7222

TELEX 678-401 TAB IRUN

3363 W. MacArthur • Santa Ana, CA 92704 • (714) 979-9923

All prices cash discounted / Freight: FOB factory. Ask for our free catalog.

M&E COMPUTRONICS INC.

•••EVERYTHING FOR YOUR TRS-80™•••

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

PAYROLL

Payroll invoices many complex calculations and the production of reports and documents, many of which are required by government agencies. It is an ideal candidate for the computer. With this Payroll system in-house, you can promptly and accurately pay your employees and generate accruate documents/reports to management, employees, and appropriate government agencies concerning earnings, taxes, and other deductions. The package has been converted to the TRS-80™ and is now a well documented, op-line, interactive, micro-computer system with the capabilities of (or exceeding) many larger systems.

CAPABILITIES:

- ★ performs all necessary payroll tasks including:
  - file maintenance, pay data entry and verification
  - computation of pay and deduction amounts
  - printing of reports and checks
- ★ can handle salaried and hourly employees
- ★ employees can receive:
  - hourly or salary wage
  - vacation pay
  - holiday pay
  - piecework pay
  - overtime pay
- ★ employees can be paid using any combination of pay types (except, hourly cannot receive salary and salary cannot receive hourly)
- ★ special non-taxable or taxable lump sums can be paid regularly or one time (bonus, reimbursements, etc)
- ★ health and welfare deductions can be automatically calculated for each employee
- ★ earnings-to-date are accumulated and added to permanent records; taxes are computed and deducted: US income tax, Social Security tax, state income tax, other deductions (regular or one time)
- ★ paychecks are printed; computer checks with your company letterhead can be purchased from SBSG
- ★ calculations are accumulated for: employee pay history, 941A report, W-2 report, insurance report, absentee report
- ★ fully linked to General Ledger. Each employee's payroll information can be distributed to as many as (12) twelve different GL accounts; system automatically posts to cash account

INVENTORY CONTROL/INVOICING

- ★ **ISAM** (Indexed Sequential Access Method) eliminates the necessity for time consuming sort.
- ★ Pre-Allocated Files for IMMEDIATE update and inquiry capabilities.
- ★ Fast Disk storage and retrieval.
- ★ Inventory Master Record includes...class...SKU...Division...Retail...Cost...Beginning Balance...Period Sale Units...Period Receipts...On Order...On Hand...Minimum Reorder Point...Recommended Reorder Amount...Vendor Number...Period Sale Dollars...YTD Sale Units...YTD Sale Dollars.
- ★ Calculated and Displayed Formulas include...Gross Margin ($)...Gross Margin (%)...Gross Margin ROI (%)...Average Inventory Retail ($)...Average Inventory Cost ($)...Turn-Over (%).
- ★ Reports Generated include...Master File Listing...Class Description Listing...Transaction Audit Trail...Minimum Reorder Point by Vendor...Retail Price List...Retail & Cost Price List...Period Sales Report ...Year to Date Sales Report...Stock Status (Screen or printer output) ...Commission Report (for salesmen and buyers).
- ★ Transaction Types include...Sales, Vendor Receipts...Vendor Orders...Customer Returns...Vendor Returns...Transfer Stock.

GENERAL LEDGER

The General Ledger accounting system consolidates financial data from other accounting subsystems (A/R, A/P, Payroll, direct posting) in an accurate and timely manner. Major reports include the Income Statement and Balance Sheet and a "special" report designed by management. The beauty of this General Ledger system is that it is completely user formatted. You "customize" the account numbers, descriptions, and report formats to suit particular business requirements. These programs were developed 5 years ago for the Wang micro-computer and have been tested in many environments since then. The package has been converted to the TRS-80™ and is now a well documented, on-line, interactive micro-computer system with the capabilities of (or exceeding) many larger systems.

CAPABILITIES:

- ★ more than 200 chart of accounts can be handled
- ★ account number structure is user defined and controlled
- ★ more than 1,750 transactions may be entered via:
  - direct posting; done by hand; validated against the account file before acceptance
  - external posting; generated by A/R, A/P, Payroll or any other user source
- ★ data is maintained and reported by:
  - month
  - quarter
  - year
  - previous three quarters
- ★ reports (samples on back) include:
  - trial balances
  - income statement
  - balance sheet
  - special accounts reports and more.....
- ★ user formats reports with the following designated as you wish:
  - titles
  - headings
  - account numbers
  - descriptions
  - subtotals
  - totals
  - skip lines
  - skip pages
- ★ up to eight levels of totals - fully user designated
- ★ menu driven; easy to use; full screen prompting and cursor control

M&E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

NEW TOLL-FREE ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

24 HOUR ORDER LINE
(914) 425-1535

PRICING	MOD-I VERSION	MOD-II VERSION
ACCOUNTS RECEIVABLE	$125	$225
ACCOUNTS PAYABLE	$125	$225
GENERAL LEDGER	$125	$225
PAYROLL	$125	$225
INVENTORY	$175	$275
INVOICING	$150	$250
COORDINATED INVENTORY/INVOICING ACCOUNTS RECEIVABLE	$449	$749
COORDINATED AR-AP-GL	$375	$675
COORDINATED AR-AP-GL with PAYROLL	$495	$899
EXTENDED COORDINATED AR-AP-GL INVOICING/INVENTORY without PAYROLL	$799	$1299

H & E COMPUTRONICS INC.

...EVERYTHING FOR YOUR TRS-80™...

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

MICROSOFT BASIC COMPILER

With TRS-80™ BASIC Compiler, your Level II programs will run at record speeds! Compiled programs execute an average of 3-10 times faster than programs run under Level II. Make extensive use of integer operations, and get speeds 20-30 times faster than the interpreter.

Best of all, BASIC Compiler does it with BASIC, the language you already know. By compiling the same source code that your current BASIC interprets, BASIC Compiler adds speed with a minimum of effort.

And you get more BASIC features to program with, since features of Microsoft's Version 5.0 BASIC interpreter are included in the package. Features like the WHILE...WEND statement, long variable names, variable length records, and the CALL statement make programming easier. An exclusive BASIC Compiler feature lets you call FORTRAN and machine language subroutines much more easily than in Level II.

Simply type in and debug your program as usual, using the BASIC interpreter. Then enter a command line telling the computer what to compile and what options to use.

Voila! Highly optimized, Z-80 machine code that your computer executes in a flash! Run it now or save it for later. Your compiled program can be saved on disk for direct execution every time.

Want to market your programs? Compiled versions are ideal for distribution. You distribute only the object code, not the source, so your genius stays fully protected.

BASIC Compiler runs on your TRS-80™ Model I with 48K and disk drive. The package includes BASIC Compiler, linking loader and BASIC library with complete documentation **$195.00**

1980 INCOME TAX PAC

Completely Revised - Latest Tax Tables - Fully Tested - Complete Manual and Documentation. The new version of the Income Tax Pacs are full of error catching codes making it impossible to make an error. Follow the simple Step By Step procedure that makes tax preparation simple.

INCOME TAX PAC A ..... **($19.95...Cassette)**

For Level II 16K Cassette Only
Does Form 1040 and 1040A
- Schedule A itemized deductions
- Schedule B interest and dividends
- Output to video display
- Schedule TC tax computation

INCOME TAX PAC B ..... **$49.95...Cassette or Diskette)**

For Level II 16K with or without printer...cassette or disk has all features of Income Tax Pac A **Plus** works with or without line printer.
- Formats Form 1040 and 1040A for standard tax forms
- Schedule C income from a personally owned business
- Form 2106 employee business expense

PROFESSIONAL INCOME TAX PAC C ..... **$99.95...Diskette**

For Level II 32K with disk and printer (optional)
Has all features of Income Tax Pac B **Plus** automatic memory storage for income tax preparers.
- 22 additional schedules and forms
- Formats forms for individual or tractor feed printing

MOD II CPA VERSION **$199.95**

GUARANTEED PROFIT **91%** **WINS PLACES SHOWS** **32%** **AVERAGE PROFIT AT ALL TRACKS-1978**

THE HORSE SELECTOR II (FLATS) (By Dr. Hal Davis **$50.00**

New simplified version of the original Horse Selector. The first Horse Selection System to actually calculate the estimated odds of each horse.

HIGHER PROFITS (OVER 100%) POSSIBLE THROUGH SELECTIVE BETTING ON:
- Rates each horse in 10 seconds.
- Easy to follow rules.
- Can be used with any Apple II Computer.
- 100% money back guarantee (returned for any reason).
- Uses 4 factors (speed rating, track variant, distance of the present race, distance of the last race).
- Using the above factors, the Horse Selector calculates the estimated odds. BET on horses whose actual payoff (from the Tote Board or Morning Lines) is higher than payoff based on estimated odds.
- Using the above factors, the Horse Selector calculates the estimated odds. BET on any selected horse with an estimated payoff (based on Tote Board or Morning Lines) higher than calculated payoff (based on Horse Selector II).
- Source listing for the TRS-80™, TI-59, HP-67, HP-41, Apple and BASIC Computers.
- No computer or calculator necessary (although a calculator would be helpful for the simple division used to calculate estimated odds).

FREE Dutching Tables allows betting on 2 or more horses with a guaranteed profit.

NEWDOS/80

A **New** enhanced NEWDOS for TRS-80™ Model I for the 1980's

Apparat Inc., announces the most powerful Disk Operating System for the TRS-80™. It has been designed for the sophisticated user and professional programmer who demands the ultimate in disk operating systems.

NEWDOS/80 is not meant to replace the present version of NEWDOS 2. 1 which satisfies most users, but is a carefully planned upward enhancement, which significantly extends NEWDOS 2. 1's capabilities. This new member to the Apparat NEWDOS' family is upward compatible with present NEWDOS 2. 1 and is supplied on Diskette, complete with enhanced NEWDOS + utility programs and documentation. Some of the NEWDOS/80 features are:
- New BASIC commands that supports with variable record lengths up to 4095 Bytes long.
- New BASIC commands that supports with variable record lengths up to 4095 Bytes long.
- Mix or match disk drives. Supports any track count from 18 to 80. Use 35, 40 or 77 track 5" mini disk drives or 8" disk drives, or any combination.
- A security boot-up for BASIC or machine code application programs. User never sees "DOSREADY" or "READY" and is unable to "BREAK", clear screen, or issue any direct BASIC statement including "LIST."
- New editing commands that allow program lines to be deleted from one location and moved to another or to allow the duplication of a program line with the deletion of the original.
- Enhanced and improved RENUMBER that allows relocation of subroutines.
- Powerful program chaining.
- Device hanging for routing to display and printer simultaneously.
- CDE function; simultaneous striking of the C, D and E keys will allow user to enter a mini-DOS to perform some DOS commands without disturbing the resident program.
- Upward compatible with NEWDOS 2. 1 and TRSDOS 2.3.
- Includes Superzap 3.0 and all Apparat 2.1 utilities.

..... **$149.00**

STOCK MARKET MONITOR

Galactic Software Ltd.

CASSETTE VERSION **$89.00**
DISK VERSION **$99.00**

1. The system is designed for the active "trader" not the "long term" investor, as the system is "technically" oriented.
2. For the TRS-80™ Model I, Level II, 16K or more. Available in both disk and tape versions.
3. Tracks user selected issues, in a technical system that reflects the issue's performance against the overall market.
4. Set up data is input by the user from the Standard and Poors stock guide or Value Line.
5. Daily issue data, "high", "low", "close" and "volume" are input from any newspaper containing this information.
6. Daily overall market, "volume" and "closing Dow" are also provided from a newspaper.
7. Volume and price changes of an issue, as they compare to volume an price changes of the overall market, are the basis of this system's analysis of the given issue.
8. Comparisons of the issue against itself are also done. This may allow the user to spot "unusual" activity on this issue.
9. Clear indications are given as to whether the issue is "out performing", "under performing" or "performing" with the market.
10. Complete video and printed output is provided.
11. This program is intended to be a guide to indications, and is not to be used as a sole recommendation to buy, sell or hold an issue. These decisions are the responsibility of the user and his brokerage.

H & E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

24 HOUR ORDER LINE
(914) 425-1535

NEW TOLL-FREE ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

★ All orders processed within 24-Hours
★ 30-Day money back guarantee on all Software (less $3 penalty for handling)

ADD $2.00 FOR SHIPPING IN UPS AREAS
ADD $3.00 FOR C.O.D. OR NON-UPS AREAS
ADD $4.00 OUTSIDE U.S.A., CANADA & MEXICO

H & E COMPUTRONICS INC.

...EVERYTHING FOR YOUR TRS-80™...

* TRS-80™ is a trademark of the Radio Shack Division of Tandy Corporation

Currently Available

MOD-II PROGRAMS

★ *All orders processed within 24-Hours*
★ *30-Day money back guarantee on all TRSDOS Software*
★ *Add $2.00 for shipping in UPS Areas*
★ *Add $3.00 for C.O.D. or NON-UPS Areas*
★ *Add $4.00 outside U.S.A., Canada & Mexico*
★ *We will match any bonafide advertised price in any of the Major Computer Magazines*

ALL SOFTWARE LISTED HERE WORKS WITH TRSDOS*

(1) SLECTRIC PENCIL (Michael Shrayer Software)... Complete word processor with extensive editing and printer formatting features.....$325 (STANDARD TRSDOS VERSION)...$350 (DIABLO, NEC OR QUME TRSDOS VERSION).

(2) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL, INVOICING AND PAYROLL (Small Business Systems Group)...an extensive business system for the serious user...can be used one module at a time or as a co-ordinated system...$225. per module...$1299 for the complete system.

(3) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL AND PAYROLL (Compumax)...a complete user oriented business system...can be used one module at a time or as a coordinated system...$140 per module... $995 for the complete system.

(4) MOD-II UTILITY PACKAGE (Racet Computes)... adds important utilities to TRSDOS copy files selectively...faster and more accurate file copying... raper bad directories displays sorted directory of all files on 1 to 4 disk drives SUPERZAP change disk ID ..and more $150.

(5) ADVENTURE #1-#9 (Scott Adams - Adventure International)...a series of games formally only available on the large computers your goal is to work your way through a maze of obstacles in order to recover a secret treasure or complete a mission...the package includes all 9 Adventures written by Scott Adams...$99.95

(6) OSF (Racet Computes)...Generalized Subroutine Facility...a series of super fast machine language utilities that can be called from a BASIC program (no machine language knowledge required)...sorts 1000 items in under 5 seconds...allows PEEK and POKE statements...move data blocks...compress and un-compress data...works under TRSDOS. $50

(7) OSM (Racet Computes)...Disk Sort Merge sorts and merges large multiple diskette files on a 1 to 4 drive system .NOT AN IN MEMORY SORT can actually alphabetize (or any other type of sort) 4 disk drives worth of data sorts one complete disk of information in 10 minutes .information is provided to use DSM with the RS MAILING PROGRAM works under TRSDOS $150.

(8) RSM (Small Systems Software)...a machine language monitor and disassembler...can be used to see and modify memory or disk sectors ..contains all the commands found on the Model-I version plus some additional commands for the MOD-II...works under TRSDOS...$39.95.

(9) BLINK BASIC LINK FACILITY (Racet Computes) Link from one BASIC program to another saving all variables chain programs without losing variables $50.

(10) BASIC CROSS REFERENCE UTILITY (Racet Computes)...lists all variables and strings used in a program (with the line numbers in which they appear) lists all GOTO's and GOSUB's (with the line numbers in which they appear) searches for any specific variables or strings (with the line number in which they appear)...$50.

(11) DEVELOPMENT PACKAGE (Racet Computes)... SUPERZAP (to see, print or change any byte on a diskette)...Disassembler and MOD-II interface to the **MICROSOFT EDITOR ASSEMBLER PLUS** including uploading services and patches for Disk I/O..assemble directly into memory save all or portions of source to disk ..dynamic debug facility (ZBUG)...extended editor commands $125.

(12) HARD/SOFT DISK SYSTEM (Racet Computes)... The software essential to interface any of the popular large hard disk drives completely compatible with your existing software and files allows up to 20 megabytes of storage (and larger)...directory expandable to handle thousands of files...$400

(13) CAMEO HARD DISK DRIVE CONTROLLER coming soon (November 1?)

(14) HARD DISK DRIVES coming soon (Nov 1?)

(15) H & E COMPUTRONICS, INC. SHARE-A-PROGRAM DISKETTE #1...works under TRSDOS...a collection of programs written by MOD-II owners... programs include data base management...a word processor...mail system...mortgage calculations checkbook register ..and many others $8 (add $3 postage outside of the United States, Canada and Mexico) .FREE if you send us a diskette containing a program that can be added to the SHARE-A-PROGRAM DISKETTE

(16) WABASH CERTIFIED DISKETTES $39.95 (per box of 10)

(17) FLIP SORT DISKETTE STORAGE TRAY...Stores 50 diskettes ..comes complete with index-dividers, tilt plates and adjustable spacing $44.95

(18) MASTER PAC 100...100 essential programs BUSINESS...PERSONAL FINANCE STATISTICS MATH...GAMBLING...GAMES...includes 125 page manual and 2 diskettes...$99.95

(19) BUSINESS PAC 100...100 essential business programs...INVENTORY CONTROL...PAYROLL... BOOKKEEPING SYSTEM STOCK CALCULATIONS...CHECKBOOK MAINTENANCE...ACCOUNTS RECEIVABLE ACCOUNTS PAYABLE... includes 125 page manual and two diskettes $149.95

(20) EDITOR ASSEMBLER (Galactic Software Ltd.)... the first user oriented Editor Assembler for the MDOEL II and was designed to utilize all the features of the MODEL II. It includes innovative features for ease of coding and debugging and complete documentation (over 120 pages)...works under TRSDOS $229.00

(21) BASIC COMPILER (Microsoft)...changes your source programs into machine language...increases program execution by 3-10 times $395.

(22) MAIL/FILE SYSTEM from Galactic Software Ltd. stores 2,500 names per disk. No sorting time is required since the file is automatically sorted by first and last name plus Zip Code on input. Retrieve by any combination of 19 user codes. Supports an 11 digit alphanumerics Zip. Supports a message line. Comes complete with user-oriented documentation (100-page manual). Allows for company name and individual of a company and complete phone number (and extension). works under TRSDOS $199.00

(23) INCOME TAX PAC Professional income tax package most forms and schedules output to video or line printer. automatic memory storage of all information...data can be added from diskette, changed and edited built in error checking...$199.95.

(24) COMPUTER GAMES (SBSG)...Mean Checker Machine, Star-Trek III, Concentration, Treasure Hunt, Banco, Dog Star Adventure...$74.95

(1) CP/M (Lifeboat Associates) ..an alternative operating system for the **MOD-II** that allows **MOD-II** owners to use any of the hundreds of programs available under **CP/M**...$170

(2) CP/M HANDBOOK...(Sybex). a step-by-step guide to **CP/M** takes the reader through each of the **CP/M** commands numberous sample programs... practical hints...reference tables $13.95

(3) GENERAL LEDGER, ACCOUNTS RECEIVABLE, ACCOUNTS PAYABLE, INVENTORY CONTROL, AND PAYROLL (Peachtree Software)...requires CP/M and MICROSOFT BASIC...professional business systems ..turn key operation ..can be used as single modules or as a coordinated system...$500 per module...$2500 for the complete system

(4) WORD-STAR...The ultimate word processor a menu driven word processing system that can be used with any printer. All standard word processing commands are included...plus many unique commands only found on WORD STAR ..requires CP/M ...$495

(5) MAIL LIST MERGE...An add on package that allows the user to send form letters (created on WORD-STAR) to any compiled mailing list (using any CP/M based MAIL program such as the PEACHTREE MAIL PROGRAM) requires CP/M, WORD STAR and andy CP/M based mail program...$150

(6) SELECTOR III (Micro-Ap) complete data management system...user defined fields and codes manages any list defined by the user...includes additional modules for simplified inventory control, accounts receivable and accounts payable...requires CBASIC-2 $295

(7) SELECTOR IV (Micro-Ap)...the ultimate data management system...all features ues the **SELECTOR III** plus...data file format conversions...full page report formatter computations...global search and replace ..hard disk compatible data/text merging $550

(8) GLECTOR (Micro-Ap) add on package to the **SELECTOR** general ledger that allows the user to define a customized chart of accounts $350

(9) CBASIC-3..a non-interactive BASIC used for many programs that run under **CP/M** allows user to make more efficient use of disk files eliminates the use of most line number references require on such programs as the **SELECTOR** $120

(10) MICROSOFT BASIC...an enhanced version of the **MICROSOFT BASIC** found on **TRSDOS** adds commands such as chaining (allows the user to LOAD and RUN a new program without losing the variables currently in memory)...long variable length file records, WHILE/WEND and others can be used with the BASIC COMPILER to speed up programs (3-10 times faster execution) $325

(11) MASTER TAX (CPAids) professional tax preparation program prepares schedules, A, B, C, D, E, F, G, R/RP, SE, TC, ES and forms 2106, 2119, 2210, 3468, 3903, 2441, 4625, 4726, 4797, 4972, 5695 and 6521. Printing can be on readily available pre-printed continuous forms, on overlays, or on computer generated IRS approved forms. Maintains client history files...interactive with CP/Aids General Ledger $995

(12) GENERAL LEDGER II (CPAids)...designed for CPA's...stores complete 12 month detailed history of transactions...generates financial statements, depreciation, loan amortizations, journals, trial balances, statements of changes in financial position, and compilation letters...includes payroll system with automating posting to general ledgers...prints payroll register, W2's and payroll checks...$450.

(13) ELECTRIC PENCIL (Michael Shrayer Software) Complete word processor with extensive editing and printer formatting features. $275 (Standard printer version). $300 (DIABLO, NEC or QUME version)

(14) BASIC COMPILER (Microsoft)...changes your source programs into machine language...increases program execution by 3-10 times $395

ALL PROGRAMS LISTED HERE REQUIRE CP/M*

* (CP/M IS A REGISTERED TRADEMARK OF DIGITAL RESEARCH)

H & E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE
50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

Reader Service—see page 274

Med Systems!

Proven Educational Software

The Human Adventure allows movement through a human body's cardiovascular system. All major organ systems are accessible and fully described by the computer. A graphic CAT-scan constantly shows the user his position in the body. The exploration mode allows simple exploration, while the game mode places the user in a race against time to cure the patient of cancer using his knowledge of the body's layout. Recommended for reading age through adult.

The Playful Professor is a mathematics learning aid that provides tutoring in integer mathematics and fractions for the four basic operations. Demonstrated solutions are completed step-by-step in a blackboard format easily understood by grade school children. Problems are presented in a game format that places the pupil in a sixty room mansion. To win, the player must catch the ghost with the key, then get to the front door before the ghost (or other player) recaptures the key. Movement is based on problem solving. Difficulty may be different for each player, allowing parents to be beaten by their children. Recommended for age 4 through adult.

Money Master tutors the young child in the use of money. The child is allowed to wander freely by paying tolls or buying objects. The tutoring screen depicts money graphically, and interactively instructs in the use of coins. This includes making payments and receiving change. New mazes are generated for each game. Graphic obstacles are randomly chosen from a library of several dozen. An average game lasts 20-30 minutes. Recommended for early readers through adult.

Each program $9.95 on cassette for TRS-80 Level II 16K, or Model III 16K. All three on diskette - $29.95, Model I only.

Graphic 3-D Adventures

These machine language programs are the first in a new breed of adventure. Instead of wandering through the English language, typing GO EAST or GO WEST, you move through a collossal maze represented on the screen three-dimensionally. Hallways recede into infinity or come to dead-ends. Doors open to left and right. As you encounter objects, monsters, and mayhem, one or two word commands may be used. The command set is extensive and sophisticated. Movement is via the arrow keys. Graphics generation is instantaneous. Mazes are bit-coded and **HUGE**. There is simply nothing like these programs on the market today.

Deathmaze 5000 places you on the top floor of a five-story building. Each floor is a maze of twisting passageways. Floors are connected by elevators and open pits. You have but one goal, **Escape Alive!** Where is the only door out of this nightmare? Monsters, bats, mad dogs, hunger, and many more horrors plague your every step as you struggle to escape the most complex adventure ever written.

Labyrinth places you in a maze of gigantic proportions. But you are not alone! A minotaur searches for you, seeking a grisly meal. You must find weapons, spells, and treasures. You must deal with ghosts and cave gnomes. You must avoid the minotaur until the moment is right for the final battle. And if this isn't enough, the Labyrinth twists space and time so that you may not know whether you are coming or going!

Each program $12.95 on cassette for TRS-80 Level II 16K, or Model III 16K. Both on diskette - $29.95, Model I only.

ATTENTION DEATHMAZE FANATICS!

Still on the first level? You would look much better wearing the hat. But don't charge the wrong wall!

Satisfaction Guaranteed! All Med Systems Software products come with a 14-day moneyback guarantee. If for any reason you are not satisfied, return you order within 14 days for a prompt and cheerful refund.

Ordering Information. Orders are processed within two working days. Mastercard and Visa card holders please remember to include the expiration date. We pay all postage and handling within the U.S., Canada, and U.S. territories. European orders please include $2.00 for air post.

☐ Human Adventure	$ 9.95	$
☐ Playful Professor	$ 9.95	$
☐ Money Master	$ 9.95	$
☐ Deathmaze 5000	$12.95	$
☐ Labyrinth	$12.95	$
☐ Educational Diskette	$29.95	$
☐ Deathmaze/Labyrinth Diskette	$29.95	$
	TOTAL	$

Name _____

Street _____

City _____ State _____ Zip _____

☐ MASTERCARD ☐ VISA ☐ Check

Mastercard or Visa # _____

Expiration Date _____

Med Systems Software

P.O. Box 2674 Department B69

Chapel Hill, North Carolina 27514

(919) 933-1990

The TRS-80 psychoanalyzed.

Mysteries of the Level II ROM

Victor Griswold
20 Fieldcrest Drive
Jackson, TN 38301

The Level II TRS-80 is an excellent microcomputer, but one does encounter a few difficulties when penetrating the secrets of the Level II reserved RAM. Several diligent software detectives have written articles that do much to show the inner workings of Level II.

I have used the following conventions in this article: Unsigned binary format for numbers stored in RAM will be assumed. Numerals with a *D* suffix or none at all will be in decimal. Hexadecimal numerals will have an *H* suffix and have leading zeros to indicate either a one- or two-byte value. The operating instructions for the subroutines and example programs are printed beside the program listings. This, I hope will make easier reading.

The information here came from inspecting the reserved RAM after running many test programs.

Keyboard

The keyboard driver is accessed by CALLing 002BH. The starting address of the KI device control block (4015H) is loaded into the Z-80 microprocessor's DE register pair, and some of the other registers are saved. This routine then jumps to the device control block handling routine, which saves the remaining registers (except for IY) and then branches to the DCB driver address. When returned from the driver, the registers are restored, with the accumulator containing the ASCII code of the entered character.

Note that the routine is always in ROM except for one special case explained later. The only way that the keyboard driver can be modified is by changing the DCB driver address bytes. Radio Shack's KBFIX and other custom drivers are new keyboard drivers which might or might not JUMP back into the ROM driver after their task is performed, even if that task would only require one byte of opcode in the ROM driver.

There is one exception to coming out of ROM: the BREAK key. Whenever BREAK is depressed, the keyboard driver performs a ReSTart to 0028H, which in turn makes the Z-80 JUMP to 16396 (400CH) in RAM.

Normally, there is a RETurn instruction in Level II or another JUMP in TRSDOS. There are three bytes available. You can easily POKE an XOR A (opcode 175D or AFH) and then a RETurn (opcode 201D or C9H) into these RAM locations so that whenever BREAK is depressed, the Z-80 accumulator is cleared before the keyboard driver is left. This effectively disables the BREAK key.

In order to provide multi-key rollover, the KI drivers saves an image of the old keyboard memory (except for the space bar) in RAM locations 16438-16444 (4036H-403CH). The first byte is the first row (lowest address), and so on. When directly scanning the keyboard from BASIC (bypassing INKEY$ for a repeat action whenever a key remains depressed), it is easier to scan this RAM area instead of the keyboard memory because rows of keys are only one byte apart.

Speaking about INKEY$, RAM location 16537 (4099H) stores the ASCII code of the most recent entered character. This byte is what INKEY$ references whenever it returns a character to a program. INKEY$ resets the byte to zero after the reference. Location 16537 can be preset by a POKE in order to have INKEY$ return a specific character unless a key is depressed.

One last note about the keyboard: every Level II TRS-80 keyboard can produce ASCII control codes easily and without any hardware or software modification. Simply depress SHIFT and the DOWN ARROW simultaneously, and then depress the appropriate letter key.

I/O Buffer

The input/output buffer is not explained in the Level II manual. It is used for program line input and output (LIST), condensing the program lines before they are put into RAM, holding the text during an INPUT statement, and for INPUT# from cassette. It is not used during CLOAD or PRINT#.

The management of the buffer is straightforward. The handler stores input characters in the buffer and puts a code zero after the last valid character for either the ENTER or BREAK terminating character. The storage of other control codes depends on the type of data manipulated at the time. RAM locations 16551-16552 (40A7H-40A8H) not only control where the handler begins the buffer, but also where BASIC starts interpreting its contents.

This means that the buffer can be positioned anywhere in nonsensitive RAM (RAM which is not used for BASIC's "housekeeping"), and BASIC won't know the difference. For instance, disk BASIC uses a different buffer than Level II. A pro-

The DATA-TRANS 1000

A completely refurbished **IBM** Selectric Terminal with built-in **ASCII** Interface.

***FOR YOUR TRS-80 WITH OR WITHOUT EXPANSION INTERFACE.**

Features:

- 300 Baud
- 14.9 characters per second printout
- Reliable heavy duty Selectric mechanism
- RS-232C Interface
- Documentation included
- 60 day warranty - parts and labor
- High quality Selectric printing Off-line use as typewriter
- Optional tractor feed available
- 15 inch carriage width

Also works with Exatron's Stringy floppy, for fast loading of programs. (Has R5232 built in stringy)

Desk and table top models also available.

HOW TO ORDER DATA-TRANS 1000

1. We accept Visa, Master Charge. Make cashiers checks or personal check payable to:

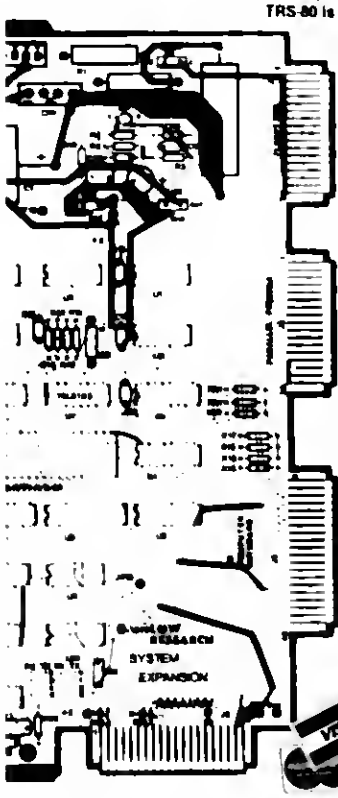
DATA-TRANS

2. All orders are shipped F.O.B. San Jose, CA
3. Deliveries are immediate

For orders and information

DATA-TRANS
2154 O'Toole St. ✓274
Unit E
San Jose, CA 95131
Phone: (408) 263-9246

TRS-80 is a registered trademark of TANDY CORP.



SYSTEM EXPANSION

FOR THE

TRS-80™

AT

$69.95 [PC BOARD & USER MANUAL]

- SERIAL RS232C 20mA I O
- FLOPPY CONTROLLER
- 32K BYTES MEMORY
- PARALLEL PRINTER PORT
- DUAL CASSETTE PORT
- REAL-TIME CLOCK
- SCREEN PRINTER BUS
- ONBOARD POWER SUPPLY
- SOFTWARE COMPATIBLE
- SOLDER MASK, SILK SCREEN

LNW RESEARCH ✓53

8 Hollowglen St. Irvine CA
714-552-8946 92714

TO ORDER

P.O. Box 16216 Irvine CA 92713
Add $3 for postage and handling.
CA residents add 6% sales tax

Card No. _____ Expiration Date _____ Signature _____

ZIP UP TO 7 SPEEDS!

RUN YOUR TRS-80 RELIABLY UP TO 2.25 TIMES FASTER (4MHZ) BY PURCHASING FROM THE ONLY MANUFACTURER OF SPEEDUP UNITS TO RECEIVE WIDE-SPREAD NATIONAL ACCLAIM FOR ITS PRODUCT. HERE'S A FEW EXAMPLES OF COMMENTS ON OUR ORIGINAL BOARD: "...elegant device ..does what it claims.", Kilobaud MICROCOMPUTING, Oct 80 • (Bill Archbold's Speedup Board *and Video I* are) "some great things", INTERFACE AGE, Jan 80 • "...a worthwhile modification for the TRS-80," 80-US Journal, Sept/Oct 79 • "...beautifully assembled...", CIE TRS-80 Bulletin, May 79. Our NEW unit has many added features • run programs 50% slower than normal, normal, and 50%, 70%, 90%, 100%, or 125% faster (a 50% minimum increase is guaranteed, 90 to 100% typical, with no additional hardware • shows changes required to the TRS-80 to insure reliable operation up to 4mhz • software control with manual override option • compensates for slow memory • power LED changes color to indicate operating speed • supports speeds far in excess of 4mhz should they prove practical in the future.

ASSEMBLED & TESTED $37.50

VIDEO I. An electronic addition that provides black characters and graphics on an all white screen for a much easier to read presentation — gives none of the glare associated with plastic screen add-ons. Software controllable. For use with TRS-80 monitors only.

ASSEMBLED $23.95

Calif. residents add 6% tax. Foreign orders add 10%.

ARCHBOLD ELECTRONICS

10708 Segovia Way Rancho Cordova, CA 95670
(916) 635-5408
Dealer inquiries invited

gram can INPUT directly into a string or into high memory from the keyboard or cassette, or even load data into video memory directly from keyboard or cassette.

Maintaining the buffer in video memory for keyboard use is not practical, since BASIC will lock up if the buffer is there while BASIC is in command mode (not due to scrolling). Using this ability for recovery of lost PRINT# cassette files, however, is another matter. The computer will eventually return control to the user after a bad cassette data block has been read, but it is not unusual for the computer to lock up or at least to clear the INPUT# variables instead of putting a value into them. This way you can at least see the data before the computer forgets it or locks up. Program Listings 1 and 2 illustrate the above points.

BASIC puts a comma in front of the first character of the buffer during an INPUT or INPUT#. It puts a colon three bytes before the first character while in command mode (statement condensing brings the condensed statements two characters before the first character). This is why the I/O buffer in the Level II memory map shows 16870 (41E6H) as the start location of the buffer, while RAM locations 16551-16552 indicate 16872.

The Video Display

The branch to the video display driver is done exactly as the branch to the keyboard, except that the CALL entry point is 0033H. There is no found exit from ROM this time. Thus, the only way to modify the driver would be to alter the DCB driver address.

RAM location 16445 (403DH) holds the 64 characters per line/32 characters per line status. A 00H at this location means 64 char./line; 08H means 32 char./line. The video driver uses this byte to determine whether to single or double-space PRINTed characters.

The BASIC Program

As most users know, Level II BASIC programs are compressed in RAM. This simply

```
10 CLS : CLEAR600
20 POKE 16551,2 : POKE16552,63  ' START THE I/O BUFFER
     3/4
   DOWN THE VIDEO DISPLAY
30 PRINT CHR$(28) :
   INPUT "TYPE SOMETHING IN" ; A$
     : PRINT A$
   ' HOME CURSOR TO PREVENT SCROLLING,
     AND INPUT STRING
40 IF LEFT$(A$,3)<>"END" THEN GOTO 30
   ELSE POKE 1655
     1,232 : POKE 16552,65
   ' IF MORE EXPERIMENTATION
     , CONTINUE.  OTHERWISE, RESTORE
   NORMAL I/O BUFF
     ER LOCATION.
50 ' NOTE THAT SINCE NORMAL VIDEO MEMORY CAN NOT RETURN
     THE
   END-OF-BUFFER 00H CODE (IT APPEARS AS AN "
     @"), BASIC USES
   A DEFAULT STRING LENGTH.
```

Program Listing 1. The above demonstretes I/O buffer operation by relocating the buffer into the video memory.

```
10 CLS
20 POKE 16551,2 : POKE16552,63  ' START I/O BUFFER 3/4

       DOWN THE VIDEO DISPLAY
30 INPUT "DEPRESS <ENTER> WHEN TAPE IS READY" ; A$ :

     INPUT#-1, A$  ' WAIT FOR OPERATOR-READY AND THEN
     INPUT
   CASSETTE DATA
40 POKE 16551,232 : POKE 16552,65   ' RESTORE NORMAL I/
     O BUFFER
   LOCATION
```

Program Listing 2. The above can be used to partially recover "unreadable" cassette data blocks.

✔ Reader Service—see page 274

means that the statements, functions and operators are represented by a one-byte code. In addition, certain pointers are set up within each line.

Tables 1 and 2 respectively show the numeric and alphabetic order listings of the compression codes. Note that each ELSE statement has an unseen colon before it, TAB is actually TAB(, and " ' " is a normal ":REM" sequence followed by the special code 251 (FBH).

Each program line takes the following form: a two-byte binary line pointer which points to the first byte of the line pointer of the next program line, a two-byte binary line number, the program line itself, and a code zero to indicate the end of the line. After the last line in a program, two zero bytes are placed where the next line's pointer would normally be. One zero byte is also placed before the first line's pointer. Among other things, this initializes data statements.

BASIC stores the names of the statements (those which you see in a LIST) in ROM locations 5712-6175 (1650H-181FH). There are no special terminator codes to separate one statement name from another in this lookup table. Rather bit 7 (the highest bit) is set to a 1 in the first character of each statement name. The statement names are in the numeric order of the statements: END first, FOR second, etc.

In order to determine the execution address of each statement and function, BASIC normally uses two other lookup tables.

There are separate tables for statements and functions, and other codes have individual routines for comparison and JUMPing. The statement table resides at locations 6178-6297 (1822H-1899H) and covers statements END (code 128D, 80H) through NEW (code 187D, BBH). The two-byte jump addresses are in the numeric order of the statements. The function table resides at locations 5640-5711 (1608H-164FH) and covers functions SGN (code 215D,D7H) through MID$ (code 250D, FAH). Again, two-byte addressed jumps are in the numeric order of the functions.

BASIC jumps directly to each arithmetic function and not to an intermediate routine. Miscellaneous codes such as ".", and "(", and all statements and functions with codes between 187 and 215 each have a separate compare and JUMP routine, some of which can be seen in ROM locations 9394-9521 (24B2H-2531H). See Table 3.

125— ----	150— TRON	175— LPRINT	200— MEM	225— COS
126— ----	151— TROFF	176— DEF	201— INKEY$	226— SIN
127— ----	152— DEFSTR	177— POKE	202— THEN	227— TAN
128— END	153— DEFINT	178— PRINT	203— NOT	228— ATN
129— FOR	154— DEFSNG	179— CONT	204— STEP	229— PEEK
130— RESET	155— DEFDBL	180— LIST	205— +	230— CVI
131— SET	156— LINE	181— LLIST	206— -	231— CVS
132— CLS	157— EDIT	182— DELETE	207— *	232— CVD
133— CMD	158— ERROR	183— AUTO	208— /	233— EOF
134— RANDOM	159— RESUME	184— CLEAR	209— ↑	234— LOC
135— NEXT	160— OUT	185— CLOAD	210— AND	235— LOF
136— DATA	161— ON	186— CSAVE	211— OR	236— MKI$
137— INPUT	162—OPEN	187— NEW	212— >	237— MKS$
138— DIM	163— FIELD	188— TAB(	213— =	238— MKD$
139— READ	164— GET	189— TO	214— <	239—CINT
140— LET	165— PUT	190— FN	215— SGN	240— CSNG
141— GOTO	166— CLOSE	191— USING	216— INT	241— CDBL
142— RUN	167— LOAD	192— VARPTR	217— ABS	242— FIX
143— IF	168— MERGE	193— USR	218— FRE	243— LEN
144— RESTORE	169— NAME	194— ERL	219— INP	244— STR$
145— GOSUB	170— KILL	195— ERR	220— POS	245— VAL
146— RETURN	171— LSET	196— STRING$	221— SQR	246— ASC
147— REM	172— RSET	197— INSTR	222— RND	247— CHR$
148— STOP	173— SAVE	198— POINT	223— LOG	248— LEFT$
149— LSE	174— SYSTEM	199— TIME$	224— EXP	249— RIGHT$
				250— MID$

Note that ELSE is formed by preceding code 149 by a code 58, an ordinary ASCII colon. An apostrophe-REMark is formed by placing a code 251 after a normal ":REM" (code 58, code 147) sequence.

Table 1. Numeric-Order Listing of Statement Compression Codes

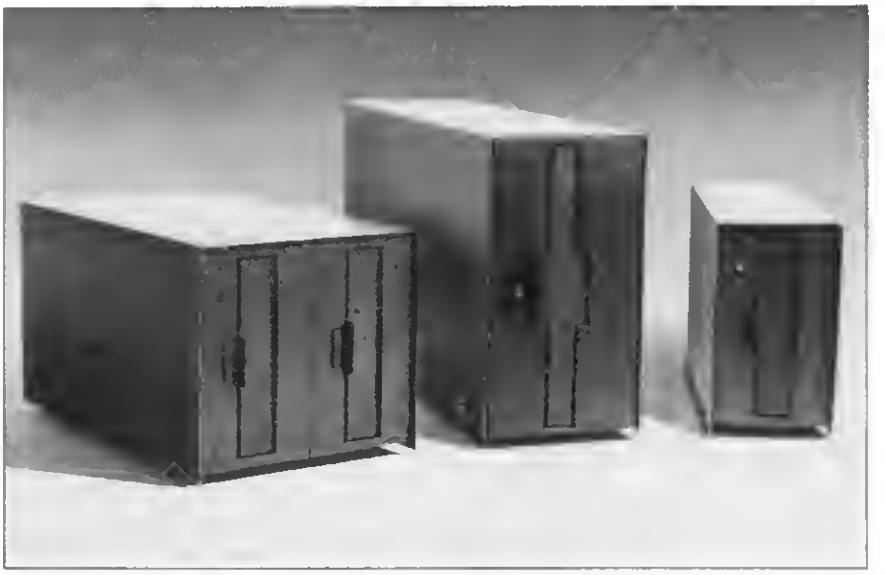
@	— 64	DEFSNG	—154	INT	—216	OPEN	—162	STEP	—204
ABS	—217	DEFSTR	—152	KILL	—170	OR	—211	STOP	—148
AND	—210	DELETE	—182	LEFT$	—248	OUT	—160	STRING$	—196
ASC	—248	DIM	—138	LEN	—243	PEEK	—229	STR$	—244
ATN	—228	EDIT	—157	LET	—140	POINT	—198	SYSTEM	—174
AUTO	—182	ELSE	—58,149	LINE	—156	POKE	—177	TAB(	—166
CDBL	—241	END	—128	LIST	—180	POS	—220	TAN	—227
CHR$	—247	EOF	—233	LLIST	—161	PRINT	—178	THEN	—202
CINT	—238	ERL	—194	LOAD	—167	PUT	—166	TIME$	—199
CLEAR	—184	ERR	—196	LOC	—234	RANDOM	—134	TO	—189
CLOAD	—185	ERROR	—158	LOF	—235	READ	—139	TROFF	—151
CLOSE	—168	EXP	—224	LOG	—223	REM	—147	TRON	—150
CLS	—132	FIELD	—183	LPRINT	—175	RESET	—130	USING	—191
CMD	—133	FIX	—242	LSET	—171	RESTORE	—144	USR	—193
CONT	—179	FN	—190	MEM	—200	RESUME	—159	VAL	—245
COS	—225	FOR	—129	MERGE	—168	RETURN	—148	VARPTR	—192
CSAVE	—186	FRE	—216	MID$	—250	RIGHT$	—249	*	—207
CSNG	—240	GET	—164	MKD$	—238	RND	—222	+	—205
CVD	—232	GOSUB	—145	MKI$	—238	RSET	—172	-	—206
CVI	—230	GOTO	—141	MKS$	—237	RUN	—142	/	—208
CVS	—231	IF	—143	NAME	—169	SAVE	—173	<	—214
DATA	—138	INKEY$	—201	NEW	—167	SET	—131	=	—213
DEF	—178	INP	—219	NEXT	—135	SGN	—215	>	—212
DEFDBL	—155	INPUT	—137	NOT	—203	SIN	—226	↑	—209
DEFINT	—153	INSTR	—197	ON	—161	SQR	—221		

Table 2. Alphabetic-Order Listing of Statement Compression Codes

Variable Storage

In Level II, the simple variables, whether integer, single or double precision, are stored directly after the BASIC program in the order in which they are first used in a program. Arrays are stored after the simple variables, again in the order in which they are first used. Consult Table 4 for descriptions of the storage formats. Notice that the variable type codes equal the number of bytes in the variable after the variable name. This allows the type number to immediately tell BASIC how many bytes there are to the next variable, thus speeding up simple variable table searches. BASIC does not need to consult a lookup table to determine variable length. It can be fooled into believing that certain variables are not in the table, if a different value for the variable's length is POKEd into the variable-type byte. See Program Listing 3.

RAM locations 16641-16666 (4101H-411AH) store BASIC's type declarations (DEFINT, etc.). The table is in alphabetical order, with each byte indicating



More power to you.

Disk drives, hardware and software— now more affordable *and* more available!

The choice is yours
Whether you need professional-looking cases and power supplies, complete disk drive packages or powerful, user-tested software, call A.M. Electronics. We manufacture and sell a complete line of affordable, high-quality and *readily-available* disk drive components and software to expand the capabilities of your TRS-80™ system.

Disk drive enclosures
All enclosures feature:

- One year power supply guarantee. Each unit is 100% tested, regulated and fused.
- Optimum venting for lower operating temperatures
- Excellent RF interference shielding. Steel covers come with lustrous dark grey finish.
- Built-in provisions for optional extender cable (5¼ only).

COMPLETE DISK DRIVE PACKAGES FROM $325!

5¼-inch drives
40-track MPI 51 w/case, power supply and extender cable **$325**
80-track MP 91 w/case, power supply and extender cable **$560**
Special Offer! 8-inch drives for Model I or II
Single Siemens drive with case and power supply **$695**
New!
Dual Siemens drives with dual case and power supply **$1240**
(90 day limited warranty on disk drives)

CASES AND POWER SUPPLIES

5½-inch enclosures
Single drive unit case and power supply **$85**
Dual drive unit case and power supply **$120**
(Extender cables are $15 each extra)

8-inch enclosures
Single drive unit case and power supply **$150**
Dual drive unit case and power supply **$250**

Attention dealers, OEM's & distributors
Having trouble finding cases and power supplies for your disk drives? Call us for details on our attractive pricing and immediate product availability.

NEW! TRS—80™ Model III
With two double-density *80 track* disk drives **$2,495**

TRS-80™ SOFTWARE PACKAGES

MAKE80©
Converts 35 or 40 track diskettes into 80 track readable diskettes **$14.95**
B-17 tape operating system
For high speed (1700 baud) loading & saving of BASIC/machine language programs, almost 4 times faster than regular TRS-80™ interface **$24.85**

SUPER UTILITY, by Kim Watt.
Stand-alone 24K machine language program for disk includes:

ZAP UTILITY
- Read/modify data regardless of disk protection • One-step track-to-track/sector to sector • Dual cursors; ASCII and Hex. Modify in Hex, Decimal or ASCII • Display disk sectors, display file sectors, copy disk sectors, compare disk sectors, display/modify main memory, search memory or disk for specified string and return location

PURGE UTILITY
- Kill files by file spec or category • Zero out unused directories or sectors
- Compute passwords, change disk's name, date, passwords, protection levels • Directory routine indicates all active and inactive files, their location in directory and status of granules

DISK COPY UTILITY
- Copy any TRS-80™ readable disk, regardless of protection

TAPE COPY UTILITY
- Copy *any* TRS-80™ readable tape, regardless of protection or baud rate

DISK REPAIR UTILITY
- *Automaticelly* repair damaged HIT, GAT or BOOT sectors • Directory check advises of errors • Automatic recovery of killed files • Shows active and inactive files, and their location on the disk

MEMORY UTILITY
- Move memory • Jump to memory • Test memory • Compare memory • Input or output any byte to any port • Zero memory • Exchange memory • Edit memory • Load memory

SUPER UTILITY is now available for **$49.95**, plus $2.50 shipping and handling!

The power behind the drives©

✓452

A.M. ELECTRONICS, INC.
3366 Washtenaw Ave.
Ann Arbor, Michigan 48104
(313) 973-2312

*TRS-80 is a trademark of Tandy Corp

Visit our retail showroom for a "hands-on" look at our wide selection of TRS-80™ hardware, software and peripherals! Hours: Tues.-Fri. 11-7, Sat. 11-5

keyword	address	keyword	address	keyword	address
END	7598D, 1DAEH	OUT	11003D, 2AFBH	INP	10991D, 2AEFH
FOR	7329D, 1CA1H	ON	8044D, 1F6CH	POS	10229D, 27F5H
RESET	3120, 0138H	OPEN	16761D, 4178H	SQR	5095D, 13E7H
SET	3090, 0135H	FIELD	16764D, 417CH	RND	5321D, 14C9H
CLS	4570, 01C9H	GET	16767D, 417FH	LOG	2057D, 0809H
CMD	16755D, 4173H	PUT	16770D, 4182H	EXP	5177D, 1439H
RANDOM	4670, 01D3H	CLOSE	16773D, 4185H	COS	5441D, 1541H
NEXT	8886D, 22B6H	LOAD	16776D, 4188H	SIN	5447D, 1547H
DATA	7941D, 1F05H	MERGE	16779D, 418BH	TAN	5544D, 15A8H
INPUT	8602D, 219AH	NAME	16782D, 418EH	ATN	5565D, 15BDH
DIM	9736D, 2608H	KILL	16785D, 4191H	PEEK	11434D, 2CAAH
READ	8687D, 21EFH	LSET	16791D, 4197H	CVI	16722D, 4152H
LET	7969D, 1F21H	RSET	16794D, 419AH	CVS	16728D, 4158H
GOTO	7674D, 1EC2H	SAVE	16800D, 41A0H	CVD	16734D, 415EH
RUN	7843D, 1EA3H	SYSTEM	690D, 02B2H	EOF	16737D, 4161H
IF	8249D, 2039H	LPRINT	8295D, 2067H	LOC	16740D, 4164H
RESTORE	7589D, 1D91H	DEF	16731D, 415BH	LOF	16743D, 4167H
GOSUB	7857D, 1E81H	POKE	11441D, 2C81H	MKI$	16746D, 416AH
RETURN	7902D, 1EOEH	PRINT	8303D, 206FH	MKS$	16749D, 416DH
REM	7943D, 1F07H	CONT	7852D, 1OE4H	MKD$	16752D, 4170H
STOP	7593D, 1DA9H	LIST	11054D, 2B2EH	CINT	2687D, 0A7FH
ELSE	7943D, 1F07H	LLIST	11049D, 2B29H	CSNG	2737D, 0AB1H
TRON	7871D, 1DF7H	DELETE	11205D, 2BC8H	CDBL	2779D, 0AD6H
TROFF	7872D, 1DF8H	AUTO	8200D, 2008H	FIX	2854D, 0B26H
DEFSTR	7880D, 1E00H	CLEAR	7902D, 1E7AH	LEN	10755D, 2A03H
DEFINT	7883D, 1E03H	CLOAD	11296D, 2C1FH	STR$	10294D, 2836H
DEFSNG	7886D, 1E06H	CSAVE	11253D, 2BF5H	VAL	10949D, 2AC5H
DEFDBL	7889D, 1E09H	NEW	6985D, 1B49H	ASC	10767D, 2A0FH
LINE	16803D, 41A3H	SGN	2442D, 098AH	CHR$	10783D, 2A1FH
EDIT	11672D, 2E60H	INT	2871D, 0B37H	LEFT$	10849D, 2A61H
ERROR	6180D, 1FF4H	ABS	2423D, 0977H	RIGHT$	10897D, 2A91H
RESUME	6111D, 1FAFH	FRE	10196D, 27D4H	MID$	10906D, 2A9AH
STRING$	10799D, 2A2FH	INSTR	16797D, 419DH	TIME$	16758D, 4176H
POINT	306D, 0132H	FN	16725D, 4155H	&	16788D, 4194H

Table 3. Jump Addresses for Statements and Functions

the default variable type by the appropriate type code (2, 3, 4 or 8). At power-up, after a RUN, or after a CLEAR, all bytes in the table contain the code for single precision, 4.

Limits of BASIC

When you enter a response to the "MEMORY SIZE?" question, BASIC stores the highest address that it is allowed to use in RAM locations 16561-16562 (40B1H-40B2H). BASIC will never use any address above that which these two bytes point to. When ENTER is used as the "MEMORY SIZE?" response, BASIC uses all available memory. However, if a number is entered in response to the question, BASIC will only use memory up to that number minus two.

The CLEAR N statement affects locations 16544-16545 (40A0H-40A1H). These locations point to the highest memory byte to be used by BASIC for non-string purposes, not the lowest byte of the string storage area. The pointer indicates the top of BASIC RAM byte (in 16561-16562) minus N, N from the CLEAR N statement.

These two pointers can be manipulated to allow a program to reserve high RAM for a POKEd machine language program or byte oriented storage area without anything having been entered in response to "MEMORY SIZE?". This high RAM area can be decreased in size at any time and expanded after any appropriate CLEAR or CLEAR N statement.

Integer Number	: byte 1—identification code 02H byte 2—second character of variable name byte 3—first character of variable name byte 4—integer value LSB byte 5—integer value MSB
Single Prec. Number	: byte 1—identification code 04H byte 2—second character of variable name byte 3—first character of variable name byte 4-7—single precision value
Double Prec. Number	: byte 1—identification code 08H byte 2—second character of variable name byte 3—first character of variable name byte 4—string length (0-255) byte 5-6—location of first character of string (LSB,MSB)
Numeric or String Array	: byte 1—appropriate identification code (2, 4, 8 or 3) byte 2—second character of variable name byte 3—first character of variable name byte 4-5—length of actual array (LOA) (LSB,MSB) byte 6—number of dimensions in array (1-255) bytes 5 + 2n – 6 + 2n—# of elements in dimension "n" (1st, 2nd, etc.) . . . bytes 7 + 2n – 5 + LOA—the array elements

The characters of the variable names are stored as the corresponding ASCII codes. If the variable name is one-letter, the "second character" byte contains a 00H. The "length of actual array" pointer indicates the number of bytes in the array which follow the pointer (# of bytes in the array after byte 5).

Table 4. Variable Storage Formats

First, CLEAR enough memory to hold your strings and reserved memory. Then POKE the value (in two-byte binary) for the new top of RAM byte into locations 16561-16562. There are now (top of RAM) minus (top of BASIC RAM) reserved bytes and (top of BASIC RAM) minus (bottom of string RAM) number of bytes of string storage. Program 3 is an illustration of this procedure.

After the initial CLEAR N and POKE's, there is the same amount of program RAM (MEM number) as if only the CLEAR N had been performed. Any subsequent CLEAR N's actually result in N bytes of string space and a corresponding amount of program RAM. BASIC would be acting normally, as if "MEMORY SIZE?" had been answered with a number. The top of BASIC RAM pointer can be raised (decreasing reserved memory) at any time with no ill effects. The top of BASIC RAM pointer can not be lowered if any strings are in the string storage area, however.

Funny things happen when BASIC is told to operate on strings in an illegal area. Note that if no strings are yet assigned a value in a program, the top of BASIC RAM pointer can be positioned in relation to a numeric variable. Because the stack begins right below it, the bottom of string RAM pointer should never be manually (via POKE) changed.

RAM locations 16548-16549 (40A4H-40A5H) store the location from which BASIC begins storing a program. These locations do much more; they indicate the byte from which BASIC lists a program, scan lines for EDIT, and start a RUN. Useful manipulation of this pointer usually requires manipulation of a few others (except for CLOAD and a couple other commands), so discussion of it will come after a discussion of those other pointers.

Variable Tables

When program variables are CLEARed, they are not really erased; specific pointers are re-

DOSPLUS 3.1

DOSPLUS VERSION 3.1 FACT SHEET

1) Variable length records.
2) Full lower case support and detection
3) Repeating keyboard w/debounce
4) Execute only protection feature for BASIC programs
5) Track support for 35-80 track drives
6) Device I/O handling (*PR, *DO, *KI)
7) Built-in screen printer
8) Multiple command chaining with "DO"
9) Built-in memory test
10) New printer driver which allows complete forms control and paging
11) Execute any DOS command from BASIC and return to BASIC
12) Free space map of diskette
13) Copy with variable length files
14) RS232 switch status and UART check
15) Create and pre-allocate files from DOS
16) Boot without re-setting clock and date
17) Display current time and date from DOS
18) New DISKDUMP sector display/modify (real-time)
19) New BACKUP (more reliable, no pack ID check)
20) New FORMAT (more reliable, no need to bulk erase disk first)

***************5 MORE UTILITIES***************

1) Single drive copy
2) Restore (dead files)
3) Purge (unwanted files)
4) Clearfile (destroy data with zeros)
5) Transfer (moves all files from one disk to another)

MICRO SYSTEMS SOFTWARE INC.
5846 Funston Street, Hollywood, FL 33023
(305) 983-3390

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP _____

PHONE _____

ACCOUNT # _____

MC ☐ VISA ☐ EXP. DATE _____

PROGRAM NAME _____

QUANTITY _____

AEROCOMP DISC DRIVES FOR TRS — 80*

• MODEL 80-1 DISC DRIVE $449.95 ea.
Single-sided, "flippy", 96TPI (80 track: single density unformated 250K bytes/side; double density unformated 500K bytes/side).

• MODEL DISC DRIVE 160-2 $599.95
Double-sided, 96TPI (160 track/80 per side: single density unformated 500K bytes, double density unformated 1 megobyte)

All models are capable of single or double density and are complete with power supply and silver enclosure. Send for information on AEROCOMP 2-and 3-drive systems available in 40 and 80 track.

The Doubler™: Percom's new proprietary double-density adapter for the TRS-80* computer.

- **Increase formatted storage capacity of your minidiskettes from 1 1/2 to almost 4 times.**
- **Use with standard 5-inch drives rated for double-density operation.**
- **The DOUBLER™ reads, writes and formats either single - or double density disks. Only $219.95.**

NEW FROM MICRO-SYSTEMS!!!

Micro-Systems Software Inc. now has double density software available for TRS-80* Model I's that are equipped with the Percom Doubler.+

First is a disk editor called "Disk Zap 2.3". This editor will work either single or double density diskettes. It is track and sector oriented, and offers total access to all parts of the disk. It has the ability to format and backup diskettes as well as editing them.

Second is our new double density DOS. DOSPLUS 3.1D, like our regular DOS, will run 35-80 track drives; but offers the increased disk storage of double density.

Disk Zap is $19.95 and DOSPLUS 3.1 or 3.1D is $99.95. To order, call or write us at the address below. Master Card and Visa welcome. Orders accompanied by a personal check will be shipped when the check clears the bank.

✓384

MICRO SYSTEMS SOFTWARE INC.
Specializing in the Tandy Line

(305) 983-3390

5846 Funston Street
Hollywood, FL 33023

* TRS - 80 is a trademark of Tandy Corp
+ Doubler is a trademark of Percom Data Corp
Now available in Spanish

LYNX makes your TRS-80 a whole new animal.

LYNX is more than a telephone coupler. LYNX is a one-piece total telephone linkage system for TRS-80 Level I and II computers, with or without expansion interface. No RS-232 required for true originate/answer direct-connect telephone operation. DOS-compatible EMTERM "smart terminal" software furnished on cassette. Already have a favorite TRS-80 program? Use it with LYNX.

With LYNX you can tap the Source or the new Compu-Serve Information Utility. Control university, business and personal computers from a remote location. Communicate via electronic mail. Learn from library data bases. Profit by instant financial market info.

All for only $279.95* at your dealer or:

esi EMTROL SYSTEMS, INC. ✓278

123 LOCUST STREET LANCASTER, PENNSYLVANIA 17602
Phone 717/291-1116

VISA or Master Card Welcome

*Add $2.50 for shipping and handling. PA residents add 6% sales tax. Includes all cables. "EMTERM" terminal program, instruction manual FCC Registration Number: A909KE-69488-DM-N

Factory orders shipped same day.

"TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation"

IF YOU LIKED SANTA PARAVIA, ENJOYED GALACTIC EMPIRE, YOU'LL JUST LOVE PROJECT OMEGA

TRS-80

NEW

A PROGRAM PAINSTAKINGLY RESEARCHED FOR A TRULY AUTHENTIC & STIMULATING SIMULATION . . . YOU'LL JUST LOVE IT!

TRS-80 16K TAPE $14.95 (Single Player)
TRS-80 32K DISK $24.95 (Multi-Player)

NOW AVAILABLE AT YOUR LOCAL COMPUTER STORE IF NOT CALL OR WRITE TO

DEALER INQUIRIES INVITED

Adventure International

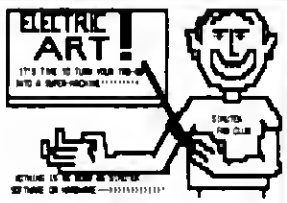
BOX 3435 LONGWOOD

OKIDATA PRINTER

The Best Printer in the World for the TRS-80! We'll Stake Our Reputation On It!

Look at the Features!

1) 200 million character head warranty! Better than any competitor!
2) Works under the most demanding business applications!
3) A "Real" 9x7 DOT Matrix Impact Printer!
4) 80 characters per second!
5) Full upper and lower case!
6) Double width characters!
7) *Supports TRS-80 Graphics! See Illustration (These graphics are exactly the same graphic codes as the TRS-80's. No special software required)*
8) Connects directly to TRS-80 with standard cable!
9) Friction & pinfeed, use roll paper, stationary or regular data paper!
10) 6 or 8 lines per inch
11) 80 and 132 columns
12) Quiet operation.



Actual photo of printout from Okidata Printer! From Simutek's Electric Artist Program!

This is the finest printer you can buy at any price for your TRS-80
Regular List Price
$850.00!
Tractor Feed Option
$150.00!

Special Simutek Customer Price Only — $559.00
(Tractor Feed $125.00 Extra)

Catalog #		
90001	Okidata Microline 80	$599.00
90002	Tractor Feed Option	125.00
91411	Cable For TRS-80 Keyboard	55.00
91401	Cable For Expansion Interface	39.00
94401	Cable For Model II TRS-80	39.00

We Accept VISA — Mastercharge — Checks — Money Orders or (C.O.D. $3.00 Extra)
NO TAX ON OUT OF STATE ORDERS! ✓19
Free Shipping In U.S.
Send Orders To: SIMUTEK, P.O. Box 13687-Z, Tucson, AZ 85732

Name _____
Address _____
City _____ State _____ Zip _____

Phone orders welcome 24 hours! (800) 528-1149 *Simutek offers other fine products for TRS-80's. Send for free catalog! Arizona residents add 4% sales tax*

TRS-80 is A TM of Radio Shack A Tandy Corp

set so that the variables can no longer be accessed. BASIC forgets the variables, just as it forgets a program after a NEW. These four pointers control the length of a program, simple variables, array variables, and data statements.

RAM locations 16633-16634 (40F9H-40FAH) indicate the top of the simple variable table, which is physically the lowest byte in the table, since that is where BASIC begins a table search. These point to the first byte of the first variable in the table, not any of the three 00H bytes at the end of a BASIC program.

Locations 16635-16636 (40FBH-40FCH) indicate the top of the array table (the lowest byte of the table). Again, these point to the first byte of the first array, not the last byte of the last simple variable. Note that if there are no simple variables, this pointer equals the top of simple variable table pointer.

Locations 16637-16638 (40FDH-40FEH) indicate the next byte after the byte at the bottom of the array table (the highest byte of the table plus one). Note that if there are no arrays, this pointer equals the top of array table pointer.

Locations 16639-16640 (40FFH-4100H) indicate the current BASIC DATA pointer. This indicates the data delimiter (comma, etc.) just before the next piece of data to be read. If the next piece of data is at the beginning of a DATA statement, the pointer indicates the 00H carriage return at the end of the program line in which the latest piece of data has been read (the 00H byte at the beginning of a BASIC program if no data has been read, or the colon after the latest DATA statement if there were other statements after that DATA statement within its program line). If the next piece of data is not right at the beginning of a particular data statement, the pointer indicates the comma preceding the piece of data—never a space (nor a semicolon, since that is not a data delimiter).

If these bytes do not point to some proper delimiter, an "OUT OF DATA" error will occur. If the pointer indicates a comma in the middle of a non-DATA statement such as a PRINT statement, BASIC will treat the PRINT statement as a DATA statement and will thus read the program as data.

The first three pointers are quite sensitive, doing anything from making a RESET necessary to causing a new "MEMORY SIZE?" if they are improperly positioned. One must organize the first three pointers in a way that seems logical to BASIC. The following are a few ways to do this.

Level II does not have an ERASE statement, which allows a program to selectively CLEAR a particular array from memory, usually so that the freed memory can be used for some other array. For example, ERASE A would eliminate array A from the table. We can't have the versatility of erasing any single array we want to, but we can erase any array if we are willing to erase all arreys below it in the array table. This means that if

```
10 A=1:B=2:C=3:D=4  ' INITIALIZE VARIABLES
20 POKE VARPTR(B)-3 , 4+7  ' POKE AN EXTRA SKIPPING-OVE
     R
   DISTANCE ( TBE REST OF B AND ALL OF C ) INTO
     B'S
   VARIABLE-TYPE BYTE
30 PRINT A,B,C,D  ' PRINT THE RESULTS
40 POKE VARPTR(A)+4 , 4  ' THE ONLY WAY TO REVERSE THE

        SKIP-OVER ( B IS NO LONGER 1,1,1, OR $ BECAUSE
     OF ITS
   I.D. )
50 PRINT A,B,C,D  ' PRINT THE RESULT
```

Program Listing 3. The above could be used when running multiple programs which share only a section of the variable table.

✓ Reader Service—see page 274

```
5 CLS
10 CLEAR:
   PRINT"PRESENT TOP OF BASIC RAM :";
      PEEK(16561)+PEEK(16562)*256:
   PRINT"PRESENT BO
     TTON OF STRING STORAGE SPACE :";
        PEEK(165
     44)+PEEK(16545)*256:
   PRINT"AMOUNT OF FREE STRIN
     G SPACE :";FRE(A$)
11 ' CLEAR OLD STRINGS AND PRINT PRESENT POINTER CONDIT
     IONS
20 INPUT "NUMBER IN 'CLEAR N' STATEMENT";N:
   CLEAR N
       ' INPUT AND CLEAR DESIRED AMOUNT OF STRING
   STO
     RAGE SPACE
30 INPUT "NEW TOP OF BASIC RAM ('MEMORY SIZE' + 2)";NS:

   POKE 16562 , INT(NS/256) : POKE 16561 , NS-INT
     (NS/256)*256
   ' INPUT AND SET NEW TOP OF BASIC R
     AM
40 PRINT : GOTO 10  ' PRINT OUT RESULTS AND RESTART EXP
     ERIMENT-
   EXIT PROGRAM VIA <BREAK> KEY.
```

Specifications
(Program 4)

- allows the programmer complete freedom to CLEAR string storage space, without losing any variables
  note: CLEARed space, of course, must not overlap the variable area (not be too large), and must not be too small to handle present string variables, if they are desired intact. The programmer may selectively save only simple variables (if an array is to be re-DIMensioned), or may save both simple variables and array variables

Usage

- save pointers:
  a. simple variables only: GOSUB 50300
  b. simple variables and arrays: GOSUB 50350
- restore pointers:
  a. simple variables only: GOSUB 50325
  b. simple variables and arrays: GOSUB 50375
- note: Read first item under "Specifications" for caution during use.
- No new variables may be introduced (nor any old variables re-introduced after the CLEAR or CLEAR N) before the pointer-restore GOSUB is made!!!
- You may, however, use in *any* way your existing old variables before the CLEAR or CLEAR N is made.

Program Listing 4. The above illustrates how to reserve high memory after the "MEMORY SIZE?" question has been answered.

the order of array initialization is A, B, C, and array B is erased, both B and C will be erased. First, compute VARPTR (first element of array, such as B(0,0)) −6−2* (the number of dimensions, in this case, 2). The 6 represents the array identification bytes, and the 2* (number of dimensions) represents the number of elements in dimension words in the array header. In a 32K RAM or 48K RAM system, the VARPTR function will return a negative value if the array is in the above 16K RAM area, and you must convert it to the actual address by adding 65536 to it. Load this number (in two-byte binary format) into the bottom of array table pointer at locations 16637-16638. If you wish to erase all the arrays and keep the simple variables, simply POKE the value of the top of array-table pointer into the bottom-of-array-table pointer.

Simple variables can be erased by repositioning the top-of-array-table pointer and setting the bottom-of-array-table pointer equal to the top-of-array-table pointer, to maintain a valid array table thereby erasing any arrays.

Program Listing 4 demonstrates the method used to CLEAR string storage space without losing variables. The old variable table pointer values are stored in memory (POKEd into the first program line) and later restored to the pointers.

By manipulating the DATA pointer a program can use multiple independent data files. Let N be the number of files and X be one of the files.

Program Listing 5 uses an N*2 BASIC array to store the

RESTORE X position and the current REENTER X position in the program's data. The RESTORE X position is set during an initialization GOSUB which tells the subroutine to scan for the beginning of the program line indicated by XP and store it in array element P(−CC,0). The REENTER X position is set to the RESTORE X position during initialization, and from then on is set to BASIC's current data pointer whenever the selected file is changed.

By manipulating several of the above pointers, the location in RAM from which BASIC begins to store a program can be changed. This involves moving the beginning of program pointer to a properly organized three-byte area in RAM and then positioning the variable table pointers to the appropriate RAM locations. A properly organized area

```
50300 '*************************************************
      ********
50301 '*       --  VARIABLE-POINTER SAVER SUBROUTINE  -
      -       *
50304 '*************************************************
      ********
50309 '
      '  SAVE SIMPLE VARIABLE POINTERS
50310 POKEPEEK(16549)*256+PEEK(16548)+7,PEEK(16635):

        POKEPEEK(16549)*256+PEEK(16548)+8,PEEK(16636)
50320 RETURN
50325 '  RESTORE SIMPLE VARIABLE POINTERS
50330 POKE16635,PEEK(PEEK(16549)*256+PEEK(16548)+7):

        POKE16636,PEEK(PEEK(16549)*256+PEEK(16548)+8)
50335 POKE16637,PEEK(16635):
      POKE16638,PEEK(16636
     )
50340 RETURN
50350 '  SAVE SIMPLE VARIABLE AND ARRAY POINTERS
50355 GOSUB50300
50360 POKEPEEK(16549)*256+PEEK(16548)+9,PEEK(16637):

        POKEPEEK(16549)*256+PEEK(16548)+10,PEEK(16638)
50370 RETURN
50375 '  RESTORE SIMPLE VARIABLE AND ARRAY POINTERS
50380 GOSUB50325
50385 POKE16637,PEEK(PEEK(16549)*256+PEEK(16548)+9):

        POKE16638,PEEK(PEEK(16549)*256+PEEK(16548)+10)
50390 RETURN
```

Specifications
(Program 5)

- allows the programmer to use multiple, independent DATA files within a program
- allows simple, fast switching between these files
- uses simple initialization procedures

Usage

- pointer initialization for file X:
  a. Array P must be DIMensioned to at least a P(X,1) size.
  b. X must be one or greater; file 0 is always initialized to start at the first line of a BASIC program.
  c. CC must indicate the value −X (must be negative).
  d. XP must indicate the line number where DATA file X begins; for line numbers greater than 32767, XP is *not* converted to "signed" format—for line number 40000, XP = 40000
  e. The program executes a GOSUB 50900.
- Note that the line-finding routine is written to conserve memory, not for speed.
- restoring the DATA pointer to the beginning of file X:
  a. CC must be 0.
  b. XP must equal X.
  c. The program executes a GOSUB 50900.
- reentry to the point in file X where READing left off:
  a. CC must be 1.
  b. XP must equal X.
  c. The program executes a GOSUB 50900.
- Note that in lines 50900 and 50940, a sign conversion is made on the PEEK addresses to allow operation on lines above the 16K RAM level—the "+ 16636*(Z1>32768)" 's can be deleted from the subroutine if no lines are above the 16K RAM level.

Program Listing 5. This subroutine, because of its nature, uses no BASIC variables. It does, however, require that the first program line in the program—line 0—be a REMark statement (REM immediately after the 0) with at least eight spaces after the REM. Line 0 is where the variable-table pointers are temporarily stored and later retrieved from by this subroutine.

Reader Service—see page 274

decimal address(es)	hex address(es)	purpose
16384-16386	4000H-4002H	RST 08H jump vector
16387-16389	4003H-4005H	RST 10H jump vector
16390-16392	4006H-4008H	RST 18H jump vector
16393-16395	4009H-400BH	RST 20H jump vector
16396-19398	400CH-400EH	RST 28H jump vector; BREAK key jump vector
16399-16401	400FH-4011H	RST 30H jump vector
16402-16404	4012H-4014H	RST 38H jump vector; interrupt mode 1 (FDC, RTC) jump vector
16438-16444	4036H-403CH	"old keyboard" image storage area
16445	403DH	video characters/line mode storage (0 = 64, 8 = 32)
16526-16527	408EH-408FH	USR JUMP transfer address
16537	4099H	most recent keyboard character
16544-16545	40A0H-40A1H	bottom of string storage area RAM minus one
16546-16547	40A2H-40A3H	current line number (line under execution)
16548-16549	40A4H-40A5H	beginning of BASIC program
16551-16552	40A7H-40A8H	I/O buffer start position
16554-16556	40AAH-40ACH	seed for random number generator
16561-16562	40B1H-40B2H	top of BASIC-usable RAM ("MEMORY SIZE?" minus two)
16598-16599	40D6H-40D7H	lowest byte of the lowest string in the string area—1
16600-16601	40D8H-40D9H	location of where BASIC is currently reading a program
16614-16615	40E6H-40E7H	location of current statement under execution
16616-16617	40E8H-40E9H	current lowest location of BASIC's stack
16620-16621	40ECH-40EDH	"." line number
16633-16634	40F9H-40FAH	top of simple variable table
16635-16636	40FBH-40FCH	top of array table
16637-16638	40FDH-40FEH	bottom of array table plus one
16639-16640	40FFH-4100H	current location of BASIC's DATA pointer
16641-16666	4101H-411AH	variable-type definition table
16722-16805	4152H-41A5H	JUMPs to DISK BASIC routines

Table 5. Numeric-Order Listing of RAM Areas

of RAM is one which makes BASIC think that the beginning of program pointer actually is at the beginning of any program, or the end of a program. In each case, the beginning of program pointer would indicate the first byte of the appropriate program line's line pointer. The simplest way to position the variable table pointers is to execute a NEW or CLOAD after positioning the program pointer.

The variable table can be repositioned anywhere in memory. One need only change each of the variable table pointers so that the simple variable teble starts at the chosen position and all other pointers are in proper relation to each other. The top array table pointer, for instance, must indicate the next byte after the last simple variable. A possible use of this is to set up a low memory byte-oriented storage area (i.e. between program and simple variables). No program lines can be added, deleted, or EDITed if the variable table is in memory lower than the program. This would cause a non-RESETable computer lockup.

The program may be changed in any way if the variable table is above the program. BASIC will not change the position of the table unless enough lines are entered so that the end of the program is beyond the start of the table. From that time on, the table will be at the true end of the program.

This ability, along with an extension of Program 4 to include the top of simple variable table pointer, would allow editing of a program without damage to the variable table.

Multiple programs can reside independently in the computer's memory at the same time. They may have the same line numbers and the same or different variable tables. Each program could call the others with some POKE statements and a GOTO statement (POKE a new beginning-of-program pointer and GOTO the desired line in the other program).

```
50980 '*************************************************
      ********
50981 '*           --  DATA STATEMENT SELECTOR  --
            *
50984 '*************************************************
      ********
50989 '
5K910 ONSGN(CC)+1GOTO5B950,5B96K
5K92K Z1=PEEK(16549)*256+PEEK(16548):P(B,B)=Z1-1:P(B,1)
     =Z1-1
50930 ZT=PEEK(Z1+3+65536*(Z1>32764))*256+
        PEEK
      (Z1+2+65536*(Z1>32765))
58948 IF2T=XPTHENP(-CC,8)=Z1-1:P(-CC,1)=Z1-1:RETURN

         ELSEZ1=PEEK(Z1+1+65536*(Z1>32766))*256

           +PEEK(Z1+65536*(Z1>32767)):
       IFZ1>BTBENG
      OTO5B93BELSEERROR5
50950 P(XP,1)=P(XP,B):IFXP=DPTBENGOTO5B965
5B96K P(DP,1)=PEEK(16648)*256+PEEK(16639)
50965 POKE16648,P(XP,1)/256:
       POKE16639,P(XP,1)-IN
      T(P(XP,1)/256)*256:
       DP=XP:RETURN
```

Variable List

Variable	Purpose
Pointer(X,F)	Pointers for each program DATA file X; Function 0 = restore to file X, Function 1 = reenter file X
XParameter	operation Parameter: during initialization, indicates line number where file starts; during restore or reenter functions, indicates the file number itself
ConditionCode	operation-type indicator
DataPointer	indicates most recent file under use
Z1	pointer to beginning of currently examined line
ZT	line number of currently examined line

Program Listing 6.

Run-Time Pointers

Three run-time pointers have some effect on error handling, STOP, and "." line update. RAM locations 16600-16601 (40D8H-40D9H) store the current position of exactly where BASIC is reading from a program. This includes command mode operation (from I/O buffer). Locations 16614-16615 (40E6H-40E7H) store the location of the beginning of the current statement under execution. This indicates either a colon for a statement in the middle of a line or the 00H carriage return of the preceding line for a statement at the beginning of a line. This pointer also operates in the command mode. The current line number is stored at locations 16546-16547 (40A2H-40A3H). This is an FFFFH for command mode.

POKEing the first two of these pointers has no visible affect on program execution because they are continually updated. POKEing an incorrect line number into the current line indicator will cause an incorrect ERL and "." line if an error occurs in the same line as the POKE.

The "." line number is stored in locations 16620-16621 (40ECH-40EDH). This is an FFFFH if there is no valid "." line.

Locations 16598-16599 (40D6H-40D7H) point to the location just below the lowest location of any string in the string storage area. This is used to assign storage positions of strings assigned a new value so that no old strings are overwritten and so that the stack is not destroyed. Note that FRE(A$) is not determined by this pointer.

Locations 16616-16617 (40E8H-40E9H) appear to point to the present lowest location of the stack pointer that is used for FOR-NEXT, GOSUB, etc.

The RND(X) seed number is stored at locations 16554-16556 (40AAH-40ACH). At power-up, all three bytes are FFH or 00H. ■

Reader Service—see page 274

Avoid
the vagueness of line numbers; use mnemonics for BASIC function calls.

Now it's Time for . . . Name That Routine

David Cornell
335 Parkside Rd.
Harrington Park, NJ 07640

BASIC is convenient, easy to learn and presents few problems when programs are small. However, as programs become large, some structural features are capable of driving one berserk. At the top of this list are line numbers.

The problem is that line numbers, while helpful in editing, give no indication of the line's function. "GOSUB 1000" can be anything. Line 47306 can be anything. Renumber the program and it's something else. Trying to understand or debug published programs is particularly maddening. One is constantly confronted by statements such as:

```
100 ON X GOSUB 1000, 2000, 3000, 40600
```

all of which can be anything.

A better system assigns a mnemonic, or label, to a routine which describes its function. The above example might then look something like this:

```
100 ON X GOSUB ADD ENTRY, FIND
    ENTRY, CHANGE ENTRY, DELETE
    ENTRY
```

Commands and Syntax

The program offered here, called Label, allows statements to be entered just this way. In the above example, "ADD ENTRY", "FIND ENTRY", etc. are names, or "labels" for routines. The label describes the function of the routine and makes it easier to both write the program, and to understand how the program functions when reading the listing.

BASIC programs created using Label will run as written, or may be used as source programs to create "object" programs in standard TRS-80 BASIC.

The commands and syntax of Label evolved from what I then considered standard BASIC programming procedures. While writing BASIC programs, I had made separate, handwritten lists of where routines were located. If a routine was to be called, its line number was looked up. If something crashed in the sort routine, its line number was looked up again, so that the routine could be found and edited. When a program was renumbered to make room for additional lines, routines had to found all over again, and a new list made.

All this was a minor inconvenience with small programs. However, programs do tend to grow (and grow and grow . . .). The result usually was a combination of scratch paper and aggravation, especially when I reminded myself that buying a computer was supposed to put an end to paperwork!

REM statements helped, of course; I began to include a REM which identified a routine. A sort routine, for example, at line 100 might be preceded by:

```
95 REM * SORT
```

The asterisk (*) was easier to spot as it zipped by on screen. But such a method was all the more annoying because keeping track of routine locations is the sort of thing that a computer can do better.

For me, the annoyance got to be too much.

I began to write a machine language program that searched the BASIC program for lines beginning with "REM*", compared strings, with a GOTO the following line if a match was found. Then I added GOSUB to the list of possibilities. Later the REM was dropped. Then I added ON, and so forth.

Now, with Label, I can call a sort routine by entering GOSUB SORT in the BASIC program. To find out where the sort routine is located, just enter FIND SORT. It's a big improvement. Label fits into low memory, thus avoiding conflict with most other machine language routines. The initialization routines then move the BASIC program above Label. This can create some problems with software that assumes BASIC will always start at the same address.

In Program Listing 1, Label is assembled at 17129, low memory on a non-disk TRS-80 system. It can, of course, be assembled to fit anywhere. The complete version, available from Instant Software, Peterborough, NH, includes a loader that calculates the lowest available memory, resolves all specific memory references, and loads the program down. All this makes Label memory independent.

Taking Control from BASIC

If TRS-80 BASIC were in RAM, this whole thing would be a lot simpler and a lot shorter. Fortunately, at various points, BASIC does make calls to reserved RAM locations. By substituting a JUMP or CALL, control can be taken from BASIC. To do this, a JUMP command is simply exchanged with the original BASIC command, saving that original command for later execution.

Label makes use of two RAM locations in this manner. In the command mode, Label examines each entry, substituting a token for each reserved word it finds. Before execution, however, a CALL is made to RAM at location 41B2. At this point a check can be made for commands recognized by Label or any user program. In fact, any number of programs can be connected, control passing from one to the other, and returning to BASIC if nothing is recognized.

At run time, and *before each*

line is interpreted by BASIC, a keyboard scan routine is called to check for BREAK or SHIFT@. This routine has a RAM call to 41C4. If the return address is the line interpreter in ROM, 1D21, control is taken from the normal BASIC process by substituting a return address in the stack.

Examples are usually more helpful than explenations. Table 1 shows a group of routines that might be used as the starting point for an address book program (possibly poorly conceived, in this case). This side-by-side comparison of the "source" and "object" programs should help clarify Label's main reason for existence.

Note that as long as Label is in place, the source program will run as written, but that the use of labels in place of line numbers makes the function of the program and the routines obvious.

Using Label

Label can also be used as a kind of operating system. It is composed of small, callable routines that combine to form blocks of functions. These in turn combine to form larger blocks. Beceuse these routines are not sequential and may be accessed at random by a machine language CALL, they can easily be incorporated into other programs or routines.

As an example, Label uses an asterisk (*) to identify BASIC lines with target labels. You may wish to write e program similar to Label that uses "%" to identify lines, or you may need to find a line that begins with a specified character. The routines in Label are available for that purpose. Consulting the listing, you will see that the routine that searches the BASIC program for a target label is called FNDKEY.

Each line of Program Listing 1 explains its own function. Examining the listing will show you how to use the Z-80 registers on entry to a routine, and what values will be returned. Table 2 defines all of Label's routines.

Syntax

An * at the start of a line identifies a label. The character string following the * is the name (*i.e.*, label) of the routine that begins on the following line of the program.

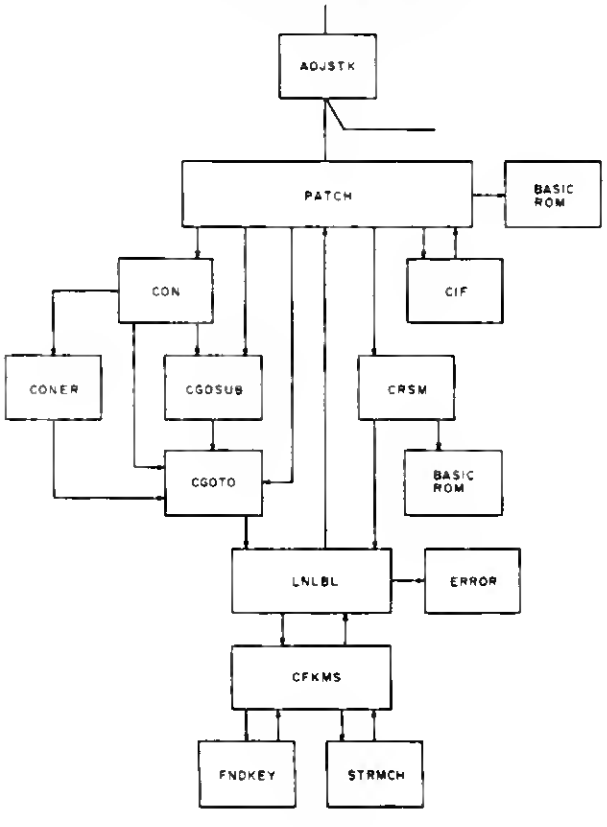


Figure 1

✓ Reader Service—see page 274

For exemple:

```
50 * MENU
60 INPUT "CMD";C$
 .
 .
100 GOSUB PUT IN ARRAY
110 GOTO MENU
 .
 .
500 * PUT IN ARRAY
510 N = N + 1
520 A(N) = X
530 RETURN
```

MENU is the name of the routine beginning at line 60; likewise, PUT IN ARRAY is the label for the routine beginning at line 510.

After entering BASIC (the command word, not the language), Label replaces all labels which are the object of GOTO, GOSUB, etc., with line numbers.

For example:

```
BASIC (enter)
50 * MENU
60 INPUT "CMD";C$
 .
 .
100 GOSUB 510
110 GOTO 60
 .
 .
500 * PUT IN ARRAY
510 N = N + 1
```

The command BASIC has two additional "switches," DELETE and REM. By entering BASIC DELETE, lines 50 and 500 (the lines with the label identifier *) are deleted.

Entering BASIC REM changes lines 50 and 500 to REM statements. The command DELETE LINE REM deletes all lines which begin with REM.

The command LIST ROUT lists all label lines, identifying all your routines:

```
LIST ROUT (enter)
50 * MENU
500 * PUT IN ARRAY
READY
>_
```

If labels are present in IF-THEN-ELSE statements, the commands GOTO and GOSUB must be used:

```
100 IF X = 1 GOTO PUT IN ARRAY ELSE
    GOTO GET FROM ARRAY
200 IF X = 2 THEN GOTO INPUT
    ADDRESS
```

The line: 100 IF X = 1 THEN PUT IN ARRAY will not work. Numbers and labels can be mixed as in the following line.

```
950 ON X GOSUB DEC/FRAC, 200,
    DISPLAY RESULT
```

While Label is operating, all lines beginning with the label identifier will be treated as REM statements, meaning they will be ignored by the BASIC program.

A valid label is any character string which does not begin with a number or delimiter. The delimiters are the colon, the comma or a zero byte (not ASCII 0). Delimiters indicate the end of the label and, obviously, may not be used as part of the label.

Labels can be any length consistent with BASIC (255 characters maximum), thus permitting detailed descriptive names to be used in your program develop-

Source Program Will Run as Written

```
10 REM THIS IS A DEMO FOR LABEL
100 CLS
110 * MENU
120  PRINT "1. NEW ENTRY"
130  PRINT "2. LIST ENTRIES"
140  PRINT
150      INPUT X: CLS
160 ON X GOSUB NEW ENTRY, LIST ENTRIES
170 GOTO MENU
180 * NEW ENTRY
190 GOSUB INPUT NAME
200 GOSUB INPUT ADDRESS
210 GOSUB APPROVE ENTRY
220      IF X$ = "Y" GOSUB PUT IN ARRAY
230 RETURN
240 * LIST ENTRIES
250 GOSUB HEADING
260 FOR E = 1 TO EN
270   GOSUB PRINT ENTRY
280   NEXT
290 RETURN
300 * INPUT NAME
310 INPUT "NAME ";NA$
320 RETURN
330 * INPUT ADDRESS
340 INPUT "ADDRESS ";AD$
350 RETURN
360 * PUT IN ARRAY
370 EN = EN + 1
360      B$(EN,0) = NA$:
         B$(EN,1) = AD$
390 RETURN
400 * HEADING
410 PRINT "NAME","ADDRESS"
420 RETURN
430 * PRINT ENTRY
440 NA$ = B$(E,0): PRINT NA$,
450 AD$ = B$(E,1): PRINT AD$
460 RETURN
470 * APPROVE ENTRY
480 CLS
490 GOSUB HEADING
500 PRINT: PRINT NA$, AD$
510 * APPROVE
520 PRINT:PRINT
530 * APPROVE 1
540 PRINT"IS THIS CORRECT Y/N"
550   X$ = INKEY$:
        IF X$ = ""THEN560
560 IF (X$ = "Y" OR X$ = "N") RETURN
        ELSE GOTO APPROVE1
570 END
```

After Command Basic Rem

```
10 REM THIS IS A DEMO FOR LABEL
100 CLS
110 REM MENU
120  PRINT "1. NEW ENTRY"
130  PRINT "2. LIST ENTRYS"
140 PRINT
150      INPUT X: CLS
160 ON X GOSUB 190,250
170 GOTO 120
180 REM NEW ENTRY
190 GOSUB 310
200 GOSUB 340
210 GOSUB 480
220   IF X$ = "Y" GOSUB 370
230 RETURN
240 REM LIST ENTRYS
250 GOSUB 410
260 FOR E = 1 TO EN
270   GOSUB 440
280   NEXT
290 RETURN
300 REM INPUT NAME
310 INPUT "NAME ";NA$
320 RETURN
330 REM INPUT ADDRESS
340 INPUT "ADDRESS ";AD$
350 RETURN
360 REM PUT IN ARRAY
370 EN = EN + 1
360      B$(EN,0) = NA$:
         B$(EN,1) = AD$
390 RETURN
400 REM HEADING
410 PRINT "NAME","ADDRESS"
420 RETURN
430 REM PRINT ENTRY
440 NA$ = B$(E,0): PRINT NA$,
450 AD$ = B$(E,1): PRINT AD$
460 RETURN
470 REM APPROVE ENTRY
480 CLS
490 GOSUB 410
500 PRINT: PRINT NA$, AD$
510 REM APPROVE
520 PRINT:PRINT
530 REM APPROVE 1
540 PRINT"IS THIS CORRECT Y/N"
550      X$ = INKEY$:
           IF X$ = ""THEN550
560   IF (X$ = "Y" OR X$ = "N") RETURN
          ELSE GOTO 540
570 END
```

After Command Basic Delete

```
10 REM THIS IS A DEMO FOR LABEL
100 CLS
120  PRINT "1. NEW ENTRY"
130  PRINT "2. LIST ENTRYS"
140 PRINT
150      INPUT X: CLS
160 ON X GOSUB 190,250
170 GOTO 120
190 GOSUB 310
200 GOSUB 340
210 GOSUB 480
220   IF X$ = "Y" GOSUB 370
230 RETURN
250 GOSUB 410
260 FOR E = 1 TO EN
270   GOSUB 440
280   NEXT
290 RETURN
310 INPUT "NAME ";NA$
320 RETURN
340 INPUT "ADDRESS ";AD$
350 RETURN
370 EN = EN + 1
360      B$(EN,0) = NA$:
         B$(EN,1) = AD$
390 RETURN
410 PRINT "NAME","ADDRESS"
420 RETURN
440 NA$ = B$(E,0): PRINT NA$,
450 AD$ = B$(E,1): PRINT AD$
460 RETURN
480 CLS
490 GOSUB 410
500 PRINT: PRINT NA$, AD$
520 PRINT:PRINT
540 PRINT"IS THIS CORRECT Y/N"
560      X$ = INKEY$:
           IF X$ = ""THEN550
560   IF (X$ = "Y" OR X$ = "N") RETURN
          ELSE GOTO 540
570 END
```

Table 1

ment. In its string matching routines, Label makes a byte-for-byte comparison, although it ignores embedded spaces.

Reserved words may be used as labels as long as the syntax is not recognized by the BASIC interpreter. For example,

```
100 GOTO NEXT
200 GOSUB RESUME
```

searches for a routine named NEXT. However, the BASIC language takes precedence in the following:

```
100 RESUME NEXT
```

This would execute as normal BASIC instead of searching for a routine called NEXT.

Remarks may follow the comma or colon delimiters in Label lines. For example:

```
500 ' SORT:THIS ROUTINE SORTS
```

This has an advantage over REM statements on other lines since the Label command LIST ROUT will display both the label and the message for each routine, and LLIST ROUT will forward the information to a line printer.

Labeled routines may be accessed from the command mode. However the first command must be one recognized by standard TRS-80 BASIC. This may merely be a colon (:).

Examples: :GOTO MENU (enter)
X = 1: GOTO MENU (enter)

PATCH

Function: Replaces part of BASIC ROM beginning at 1D1E
Checks for commands which may take a label and jumps to appropriate routine
If BASIC line begins with an asterisk, process as a REM statement
On Entry: A = result of keyboard scan
On Entry and Exit to Special Case Check: A = BASIC Token – 80H
HL points to token in BASIC line

ADJSTK (ADJust STacK)

Function: Change return address to take control from BASIC
On Entry: Top of stack is first return address
Next two-byte word in stack is ultimate return and is return address to be checked.

CHECK

Function: Checks BASIC program for commands which may take a label
Calls appropriate processing routine
Writes number for labels

FNDKEY (FIND KEY)

Function: Searches BASIC program for lines beginning with search key
On Entry: Search key in register C
DE points to line where search is to begin.
On Exit: A = 0 if the end of the program was reached
IX points to the line with the label identifier and object label
HL points to the address of the object key.
DE points to the next line, the object routine.
C = Search key

MCHSTR (MatCH STRings)

Function: Compares two strings byte for byte, ignoring imbedded spaces until one of three delimiters, colon, comma on zero is reached
On Entry The address of the source string is in the two bytes of memory reserved by Label for the purpose: SRLBPT (SouRce LaBel PoinTer).
DE points to the object string.
On Exit: Carry set if match is found

PRCSGG (ProCeSs Goto/Gosub)

Function. Change labels to line numbers in BASIC program
On Entry: HL points to byte before first character of label or line number
On Exit: HL points to next byte to be checked (by CHECK) for GOTO, GOSUB, etc.

PRCSON (PRoCeSs ON)

Function: As PRCSGG but must point to each label or line number in succession until end of line is reached.

CGOSUB (Command GOSUB)
CON (Command ON)
CGOTO (Command GOTO)
CRSM (Command ReSuMe)

Function: Jump to Label if required
On Entry: HL points to command token in BASIC line.
On Exit: HL points to first character source label or line number (not applicable for all cases of RESUME).

SPACE

Function: Move part of BASIC program in RAM to allow a line number to be written in place of a label
On Entry: The address of the source label is in SRLBPT.
The length of the string to be written in the BASIC program (the ASCII representation of the line number) in LENDEC (LENgth DECimal number)
On Exit: Program is adjusted. Carry set if new line pointers must be written.

LINEIN

Function: LB ERROR check
Entered if BASIC program line entered
On Entry: BASIC line in input buffer
Line number in DE'
Note: If match is found and line numbers are the same, line was being edited.

LENSTR (LENgth STRing)

Function: Count length of a character string to delimiter
On Entry: HL points to string.
On Exit: B = Length of string

JMPR (JuMPeR)

Function: To jump to routine corresponding to Label Command number
On Entry: Register C equals the number of the word in the WORD LIST matching entry in input buffer. Zero if there is no match
On Exit: To specified routine

JMPTBL (JuMPTaBLe)

Function: Holds location of routines corresponding to words in WRDLST (Word List)

WRDLST (WoRDLIST)

Function: Holds list of commands recognized by Label (The list looks a little weird because some of the words hold imbedded BASIC tokens. Thus, the word ROUT is indicated by ASCII R followed by the token for OUT.)

WORD LIST Structure:

00	Begin List
W	
O	
R	
D	
00	End Word

RST 10H

Label uses two RST's (Restarts) in BASIC ROM.
The RST's are calls: RST 18H is equivalent to CALL 18H.

Function: Next byte to A; Spaces are ignored.
On Entry: HL points to previous byte.
On Exit: HL points to byte.
Byte in A
Z is set if (HL) is end of statement – comma, zero or colon.
C is set if (HL) is a number.

RST 18H

Function: Compare HL and DE
On Exit: Z set if equal
C set if HL<DE
HL and DE unchanged

Table 2. Assembler Mnemonics, Names and Functions of Label Routines

NEVER UNDERSOLD!

that's right, if you can find a lower price in this magazine for any of the items listed in this ad, we will reduce our price below our competitor's price. See each box below to determine how much EXTRA we will cut off of THEIR price if we're not lowest. Please consider the competitor's shipping charges. OUR SHIPPING IS FREE!*

FLOPPY DISKETTES & SUPPLIES

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE ITEMS, DEDUCT $.50 FROM OUR COMPETITOR'S PRICE. THAT'S OUR PRICE!

Call For Quantity Discounts

Verbatim Diskettes (box of 10)

5¼" MD525-01 soft, 10 or 16	$26.50
5¼" MD577-01 quad soft, 10 or 16	$33.00
8" FD34-1000 soft	$30.00
8" FD32-1000 hard	$30.00
8" FD34-8000 double density soft	$44.00
8" FD32-8000 double density hard	$44.00

Printwheels (specify style)

Qume or Diablo	$6.50

Labels

3½"x15/16" (5000 labels)	$18.75
Other sizes and quantities	CALL

Ribbons

Diablo Hy Type I	$4.95
Diablo Hy Type II	$5.25
Qume Sprint	$3.50
Centronics Zip Pack	$3.95
MANY OTHERS	CALL

PRINTERS

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE PRINTERS, DEDUCT $10 FROM OUR COMPETITOR'S PRICE, THAT'S OUR PRICE!

Paper Tiger IDS-440G	$939
Paper Tiger IDS-460G	$1193
Anadex DP-8000	$855
Anadex DP-9500	$1395
TI-810 Basic	$1625
Centronics 737	$825
NEC 5500 D w/ Bidrctnl Board	$2695
NEC 5530	$2595
VISTA Daisey Wheel Printer	$1834
Qume 5/45,5/55	CALL
Escon IBM Interface	$595

Call For Other Printers

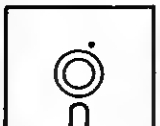
HARDWARE

IF YOU CAN FIND A LOWER PRICE IN THIS MAGAZINE ON ANY OF THESE ITEMS, DEDUCT 5% FROM OUR COMPETITOR'S PRICE, THAT'S OUR PRICE!

Novation CAT Modem	$145
Novation D-CAT Modem	$185
16K Memory kit	$46
Isolators	$49
Shugart 35tr Drive	$349
Pertec or MPI 40tr Drive	$359
Lobo Drives	CALL
Matchless Drives	CALL
Percom Doubler	$209
Percom Seperator	$27
AIM-65 Computer	375
TI-99/4 Computer	$925
California Computer Systems 8ds	CALL
Symtec Computer Boards	CALL
Mountain Hardware Boards	CALL
Green Screen	$11

Call for Other Hardware

4636 Park Granada
Calabasas, Ca. 91302



Alpha Byte Storage

For phone orders CALL:
(213) 883-8594

SPECIAL #1
If you purchase the "TRS-80 DISK AND OTHER MYSTERIES" Book for the regular price of $22.50 you can buy 10 VERBATIM DISKETTES AND a plastic library case for $22.00
TOTAL $44.50

SPECIAL #2
If you purchase APPARAT NEWDOS+ for the regular price of $99.95 you can buy 10 VERBATIM DISKETTES AND a plastic library case for $5.00
TOTAL $104.95

SPECIAL #3
If you purchase APPARAT NEWDOS/80 for the regular price of $149.00 we will give you 10 VERBATIM DISKETTES AND a plastic library case FREE
TOTAL $149.00

SPECIAL #4
If you purchase the MICROSOFT BASIC COMPILER for the REDUCED PRICE of $190.00 we will give you 10 VERBATIM DISKETTES AND a plastic library case FREE
TOTAL $190.00

SPECIAL #5
If you purchase the new "MICROSOFT BASIC REVEALED" book for the regular price of $29.95 you can buy 10 VERBATIM DISKETTES AND a plastic library case for $20.00
TOTAL $49.95

SPECIAL #6
SOFTWARE DEALS--If you want to purchase any software [including Utilities, Operating Systems, Games, Buisness Programs, etc.] we will automatically deduct from 10% to 40% off the regular retail price.

*FREE shipping on all orders over $20. Visa and Master Card accepted. All never undersold offers good as supply lasts. Please add 2.00 for all COD orders. Please call for items not listed. We gladly answer any questions on all of our hardware, software, and supply needs. Quantity discounts available. School purchase orders accepted. Please remember to figure competitors shipping and handling charges when arriving at never undersold price.

This will not work:

```
GOTO MENU (enter)
```

Three Function Blocks

Label may be considered to consist of three main function blocks: The Initialization Block (Fig. 1), which sets up the jumps and exchanges with BASIC RAM; the Run Time Block (Fig. 2), and the Label to BASIC Block (Fig. 3).

The Run Time Block consists of those routines needed when the BASIC program is running. It is further divided into three main sub-blocks which take control from normal BASIC, check for commands that may take a label, and process those com-

Run Time Block Commands

LIST ROUT	LISTs all routine lines with the label identifier.
LLIST ROUT	Prints all lines with the label identifier.
FIND XXX	Finds and displays the line and label for routine named XXX.
DELETE LINE REM	Deletes REM lines.
Label to BASIC Block Commands	
BASIC	Converts to standard BASIC. Changes label to line number following GOTO, GOSUB, etc.
BASIC DELETE	Command BASIC and deletes all lines that begin with the label identifier.
BASIC REM	Command BASIC and converts lines with the label identifier to REM statements.

Note: A comparison is made only for the length of the command, so that LIST ROUTINES produces the same result as LIST ROUT. The longer commands may be easier to remember.

Table 3. Run Time Block Commands and Label to BASIC Commands

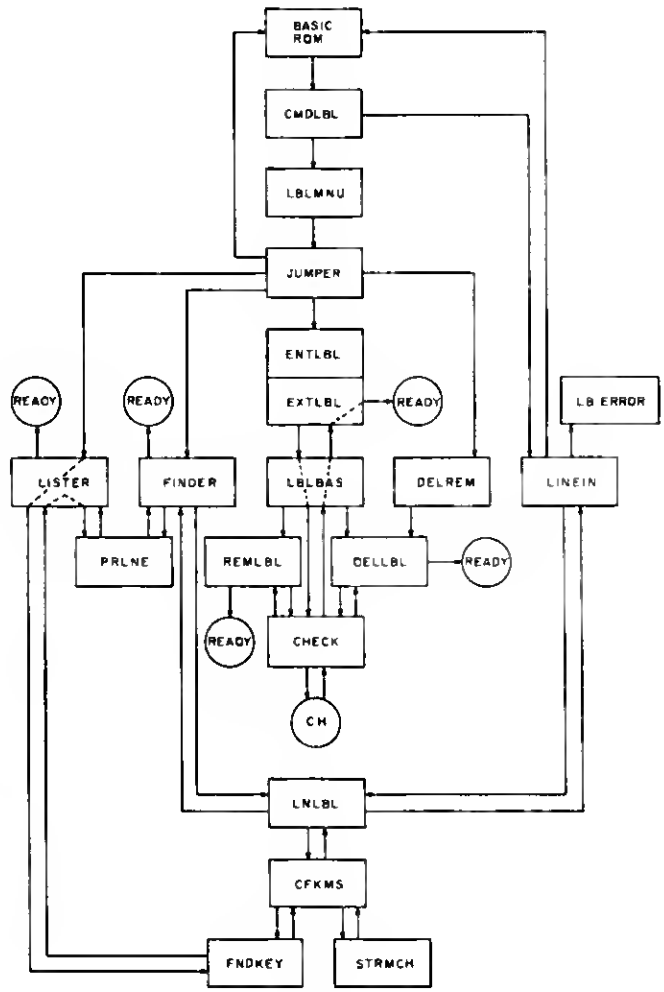


Figure 2

mands.

When a GOTO, GOSUB, or similar BASIC branching command is encountered, the next character is examined. If a number is found, a jump is made back into BASIC ROM, where the program continues normally. If a number is not found, the Label program assumes that a label has been used, and the command is processed by the appropriate Label routine.

The Label to BASIC Block is

40A4	Points to start of BASIC program
40A2	Saves line number
40A7	Pointer to BASIC input buffer
40F9	Pointer to start of variable table (scalar pointer)
40E8	Saves HL
40F0	Address of routine to be called on error
1D1E	Start line interpreter
1D9B	Scan for SHIFT @ and BREAK
0358	Calls keyboard scan routine
1AFC	Writes line pointers beginning with line pointed to by DE
1AF8	Writes line pointers from beginning of BASIC program
1A19	Ready routine
1997	SN ERROR routine
19A2	ERROR routine
1B4A	NEW routine
1B5D	BASIC initialization routines (RUN)
032A	Display register A
0FAF	Display number in HL
196C	Check for enough RAM for stack operation
2B7E	Write line of BASIC in buffer, change tokens to words
2B75	Display a string from (HL) until (HL) = 0
41B2	First RAM called after immediate mode entry
41C4	RAM called by keyscan routine 0358
4003	Saves length ASCII representation of binary integer
40D4	Points to buffer where ASCII decimal representation written

Table 4. BASIC RAM and ROM Locations Used by Label

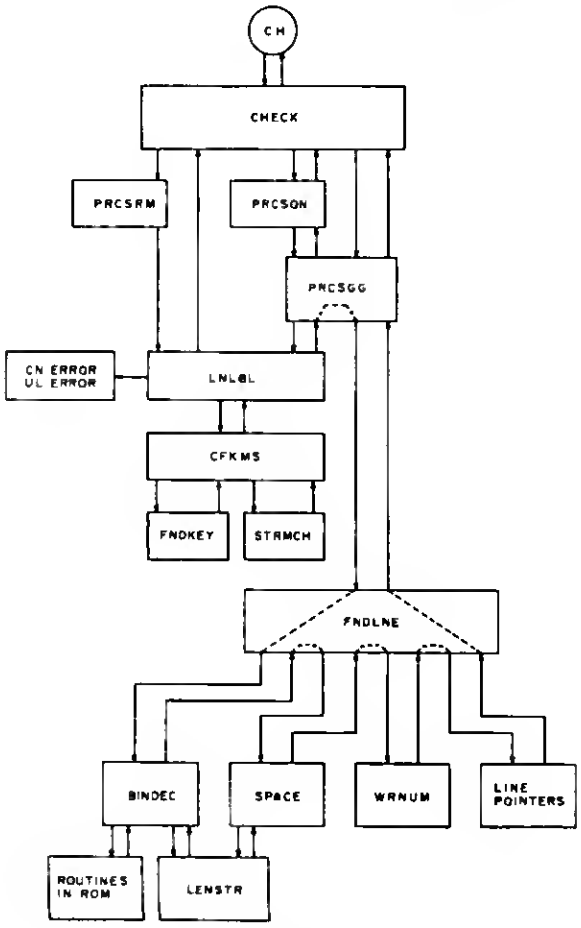


Figure 3

✓ Reader Service—see page 274

called when there is a keyboard input in the command mode. If a line of a BASIC program is being entered, and that line contains an asterisk *identifying a label*, the BASIC program is checked to see if that label has already been assigned to a routine.

Should you attempt to use a label that is already being used in the BASIC program, then the error message ERROR, LABEL ASSIGNED will be displayed. Author's comment: For once the preferred version is shorter!

If a BASIC program line is not being entered at the command level, then a check is made to see if the input is a command recognized by Label. If that is the case, a jump is made to an appropriate processing routine within Label. If the command is not recognized, then control passes on to BASIC.

The principal function of this block is to convert the source program written with labels to a standard BASIC program. The BASIC program is searched for labels; the length of the program lines is adjusted to accommodate numbers in place of labels. The numbers are calculated and written into the lines, and the various user options are executed.

Label is concerned with the commands GOTO, GOSUB, ON, ON ERROR, RESUME and IF. With the exception of IF, these commands are all variants of GOTO; RESUME is a special case of GOTO.

In both the Run Time Block and the Label to BASIC Block, then, the processing routines for these four related commands end up at the Label routine that processes the GOTO command. This centralization, in addition to making the program easier to understand, makes modification much simpler.

Table 3 explains the commands for the Run Time Block and the Label to BASIC Block. Table 4 gives the RAM and ROM locations used by Label.

User Option

Three zero bytes (indicated by NOPs) are designated USROPT (for user option) in the assembly listing. These are provided so that the user's own machine language routines can be patched in, via CALL or JP, without affecting Label.

On entry to USROPT, the new BASIC line has just been entered, and the keyboard has been scanned for BREAK or SHIFT@. USROPT is at Label ORG + 38 hex or 56 decimal. ■

Program Listing 1

```
                00001 ;*********************************
                00051 ;*   LABEL - FOR THE TRS-80      *
                00052 ;*********************************
                00053
42E9            00100          ORG     42E9H
1A19            00150 @READY   EQU     1A19H           ;ADD.READY ROUTINE IN ROM
19A2            00200 ERROR    EQU     19A2H           ;ADD.ERROR ROUTINE IN ROM
00D8            00250 GTOTOK   EQU     0D8             ;GOTO TOKEN
0091            00300 GSBTOK   EQU     91H             ;GOSUB TOKEN
00A1            00350 ONTOK    EQU     0A1H            ;ON TOKEN
40A4            00400 BPRPTR   EQU     40A4H           ;BASIC PROGRAM POINTER
184D            00450 NEW      EQU     184DH           ;ADD.NEW ROUTINE IN ROM
40A2            00500 LNNBUF   EQU     40A2H           ;LINE NUMBER BUFFER
40A7            00550 BUFPTR   EQU     40A7H           ;POINTER TO INPUT BUFFER
1997            00600 SNERR    EQU     1997H           ;SN ERROR ROUTINE IN ROM
41C4            00650 BLNRAM   EQU     41C4H           ;CALLED AT BEG.OF EACH LINE
00BF            00700 IFTOK    EQU     0FBH            ;IF TOKEN
0099            00750 ERRTOK   EQU     99H             ;ERROR TOKEN
0093            00800 RENTOK   EQU     93H             ;REN TOKEN
1AF8            00850 FWRDPT   EQU     1AF8H           ;RDOUT.WRITES LINE POINTERS
185D            00900 BASINI   EQU     185DH           ;INITIALIZATION ROUTINE
40D3            00950 LENDEC   EQU     40D3H           ;SAVES LENGTH DECIMAL #
40D4            01000 DECPTR   EQU     40D4H           ;POINTER TO DECIMAL NUMBER
40F9            01050 SCLNPT   EQU     40F9H           ;POINTER TO START OF SCALARS(VARIABLES)
009C            01100 LNETOK   EQU     09CH            ;LINE TOKEN
0086            01150 DELTOK   EQU     086H            ;DELETE TOKEN
0084            01200 LSTTOK   EQU     084H            ;LIST TOKEN
0085            01250 LLSTOK   EQU     085H            ;LLIST TOKEN
00A0            01300 OUTTOK   EQU     0A0H            ;OUT TOKEN
009F            01350 RSMTOK   EQU     9FH             ;RESUME TOKEN
0087            01400 NXTTOK   EQU     87H             ;NEXT TOKEN
4182            01450 FSTRAM   EQU     4182H           ;FIRST RAM CALLED AFTER USER INPUT
1ED9            01500 ULERR    EQU     1ED9H           ;UL ERROR ROUTINE
00CF            01550 LBIDNT   EQU     0CFH            ;LABEL IDENTIFIER
28A7            01600 DISSTR   EQU     28A7H    ;DISPLAY STRING UNTIL BYTE=0
                01650
                01700 ;ADJUST STACK: CHANGES RETURN ADDRESS TO TAKE CONTROL FROM BASIC
                01750 ;ENTRY: SECOND 2 BYTE WORD IN STACK IS ADDRESS IN QUESTION
42E9 D9         01800 ADJSTK   EXX                     ;SAVE REG'S
42EA C1         01850          POP     BC
42EB D1         01900          POP     DE
42EC 21211D     01950          LD      HL,1D21H        ;ADD.LINE INTERP.
42EF DF         02000          RST     18H             ;RET.ADD.LINE INTERP?
42F0 2003       02050          JR      NZ,KPSTK        ;NO, KEEP STACK AS IS
42F2 11FE42     02100          LD      DE,PATCH        ;YES, CHANGE RET.ADD.
42F5 D5         02150 KPSTK    PUSH    DE              ;RESTORE RETURN ADD.
42F6 C5         02200          PUSH    BC              ;RESTORE RETURN ADD.
42F7 D9         02250          EXX                     ;RESTORE REG'S
42F8 C3E942     02300 EXBLN    JP      ADJSTK          ;EXCHANGED WITH RAM DURING INIT
                02350
42FB CD5B03     02400 @IDLE    CALL    035BH           ;KEY BOARD SCAN
                02450
                02500 ;PATCH: REPLACES PART OF BASIC BEGINNING AT IDLE
                02550 ;ENTRY: A=RESULT OF KEYBOARD SCAN
42FE B7         02600 PATCH    OR      A               ;KEY BOARD INPUT?
42FF C4A01D     02650          CALL    NZ,1DA0H        ;YES
4302 22E640     02700          LD      (40E6H),HL      ;SAVE HL
```

Program continues

★★★ A PERCOM BULLETIN ★★★ ✓409

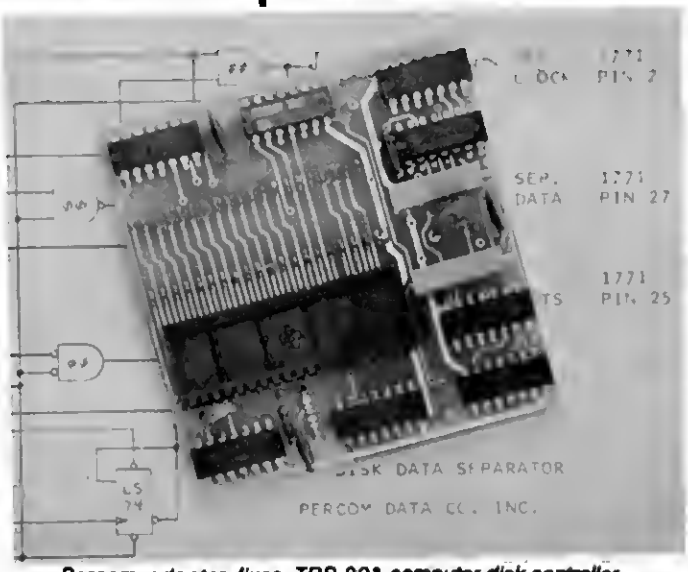
Adapter for TRS-80* computer eliminates disk read errors

Garland, Texas — Harold Mauch, president ol Percom Data Company, announced that the company is marketing a simple plug-in adapter for TRS-80* computers that corrects a design deficiency in the disk controller circuit.

The problem, which causes disk read errors, has been traced to Tandy's reliance on a circuit internal to the FD1771 controller IC to perform the function of separating clock and data pulses.

As explained in the *Backgrounder,* use of the internal chip circuit for reliable data-clock separation is a design shortcut which the manufacturer of the controller IC warns against.

The Percom solution, a PC card adapter called the SEPARATOR™, elliminates the problem by substituting an explicit data separator circuit — one which has been used reliably in Percom disk controllers since 1977 — for the internal IC separator circuit.



Percom adapter fixes TRS-80 computer disk controller.*

The SEPARATOR™ is installed without modifying the host system. The user merely removes the FD1771 IC from the host controller, installs the IC in the DIP socket on the SEPARATOR™ card, and plugs the adapter into the vacated socket of the host controller.

Percom cautions that opening the Expansion Interface of the TRS-80* computer, which is required to install the SEPARATOR™, may void the computer's limited 90-day warranty.

The SEPARATOR™, which sells for $29.95, may be purchased from Percom dealers or ordered direct from the factory. The Percom toll-free order number is 1-800-527-1592.

Payment for mail orders may be made by certified check, cashier's check or money order, or charged to a Master Card or VISA account. Texas residents must add 5% sales tax.

Percom Mini-Disk Drives Store More, Cost Less. ✓408

Percom mini-disk drives store more data, are more reliable, yet a 40-track Percom drive costs **$100.00 less** than a 35-track Tandy drive.

You can store over 102 Kbytes per disk on Percom TFD-100™ 40-track drives, over 197 Kbytes per disk on TFD-200™ 77-track drives. A patch — supplied free on minidiskette — upgrades TRSDOS* for operation with the newer 40- and 77-track drives.

Both TFD-100™ and TFD-200™ models are available in one-, two- and three-drive configurations.

Prices start at $399 for a single-drive TFD-100™, $675 for a single-drive TFD-200™. Drives are supplied with heavy-duty power supplies. Metal enclosure is finished in compatible silver enamel.

See your nearby Percom dealer or order direct by calling toll-free 1-800-527-1592.

Five-Inch Disks Store More Than Eight-Inch Disks! ✓1

Garland, Texas — June 25, 1980 — Percom Data Company has begun production of a double-density disk controller adapter for TRS-80* Model I computers.

Harold Mauch, president of Percom, made that announcement here today, saying that data storage capacity using the adapter and double-density disk operating system — which is included — can be increased to as much as 354 Kbytes per minidiskette.

By comparison, the maximum storage for larger eight-inch disk systems used with the TRS-80* Model I computer is about 290 Kbytes.

Mauch said the PC card adapter, which plugs into the controller chip socket of the computer Expansion Interface, works equally well for either single-density or double-density storage, and users may continue to run programs under TRSDOS*, OS-80™ and other single-density operating systems with the adapter installed.

Price, for the plug-in adapter, the TRSDOS*-like double-density DOS and a utility for converting files and programs from single- to double-density format is $219.95.

BACKGROUNDER

CRC ERROR! TRACK LOCKED OUT!

by the Technical Staff
Percom Data Company ✓410

This problem started while we were studying an annoying problem with the TRS-80* computer. Disk drives sold by Percom are realigned and tested before shipment. We noticed, however, that some disk drives would pass the Percom inspection but just would not work reliably on the inner tracks with a TRS-80* computer. These drives were within the manufacturer's specifications, and would function perfectly on other disk systems Percom manufactures — "perfectly" here meaning more than 50 million bytes read without error!

The disk read data separation arrangement in the TRS-80* computer Expension Interface uses an internal data separator of the FD1771 disk formatter/controller IC. Use of the FD1771 internal data separator is not recommended by Western Digital, the IC manufacturer. The following note appears on page 17 of the FD1771 data sheet:

> Internal date seperetion may work for some applications. However, for applications requiring high date recovery reliebility, WDC recommends external data seperetion be used.

We suspected the data separator because the problem was most severe on disk inner tracks where storage dansity is highest and data separation is most critical.

To prove our point, a technician breedboarded a standard Percom data separator circuit, and configured it to plug directly into the FD1771 IC socket of the TRS-80* computer controller.

When connected to the TRS-80* computer, a troublesome drive functioned perfectly! We ran a BACKUP utility many times and never got a track lockout. Before we added the external data separator circuit to the computer, this same drive would always lock out tracks, and would have difficulty reading from the inner (higher number) tracks.

The Percom data separator circuit fixes the mini-disk controller of the TRS-80* computer. The type of drives being used is irrelevant; the circuit eliminates disk read errors resulting from the inability of the Tandy controller design to reliably separate clock and data signals when reading high density inner tracks.

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.

✓258 PERCOM DATA COMPANY, INC. 211 N. Kirby Street Garland, Texas 75042 (214) 272-3421

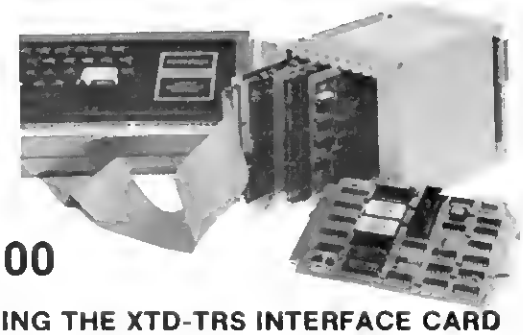
™ trademark of Percom Data Company, Inc. *trademark of Tandy Radio Shack Corporation which has no relationship to Percom Data Company.

```
4305 ED73E840 02750          LD      (40E8H),SP      ;SAVE SP
4309 78       02800          LD      A,(BL)          ;BYTE TO A
430A FE3A     02850          CP      ':'             ;END OF STATEMENT?
430C 282C     02900          JR      Z,81D5A         ;YES
430E B7       02950          OR      A               ;END OF LINE?
430F C29719   03000          JP      NZ,SNERR        ;NO, SN ERROR
4312 23       03050          INC     HL              ;END
4313 7E       03100          LD      A,(HL)          ;  OF
4314 23       03150          INC     HL              ;    PROGRAM
4315 B6       03200          OR      (HL)            ;      ?
4316 CA7E19   03250          JP      Z,197EH         ;YES
4319 23       03300          INC     HL              ;LINE
431A 5E       03350          LD      E,(HL)          ;  NUMBER
431B 23       03400          INC     HL              ;    TO
431C 56       03450          LD      D,(HL)          ;      DE
431D EB       03500          EX      DE,HL           ;SAVE
431E 22A240   03550          LD      (LNNBUF),HL     ;  LINE NUMBER
4321 00       03600 USROPT   NOP     ;*** A JUMP OR CALL TO A USER
4322 00       03650          NOP     ;* ROUTINE MAY BE PLACED HERE
4323 00       03700          NOP     ;* WITHOUT AFFECTING LABEL
4324 3A1B41   03750          LD      A,(411BH)       ;TROM/TROFF ROUTINE-
4327 B7       03800          OR      A               ;  ?
4328 280F     03850          JR      Z,81D59         ;NO
432A D5       03900          PUSH    DE
432B 3E3C     03950          LD      A,'<'
432D CD2A03   04000          CALL    032AH
4330 CDAF0F   04050          CALL    0FAFH
4333 3E3E     04100          LD      A,'>'
4335 CD2A03   04150          CALL    032AH
4338 DI       04200          POP     DE
4339 EB       04250 81D59    EX      DE,HL
433A D7       04300 81D5A    RST     10H             ;NEXT BYTE TO A
433B 11F842   04350          LD      DE,81D1E        ;RETURN ADDRESS
433E D5       04400          PUSH    DE              ;  TO STACK
433F C8       04450 81D5F    RET     Z               ;END STATEMENT
4340 FECF     04500          CP      LBIDNT          ;LABEL IDENTIFIER?
4342 CA071F   04550          JP      Z,1F07H         ;YES, NEXT LINE
4345 D600     04600 B1D60    SUB     00N
4347 DA211F   04650          JP      C,1F21H         ;BYTE IS VARIABLE
434A FE3C     04700          CP      3CH             ;TOKEN?
434C D2E72A   04750          JP      NC,2AE7H        ;NO
              04800 ; CHECK FOR SPECIAL CASES
              04850 ;ENTRY: A=BASIC TOKEN-80H; HL POINTS TO TOKEN IN BASIC LINE
434F FE21     04900          CP      ONTOK-80H       ;IS COMMAND 'ON'?
4351 2865     04950          JR      Z,CON           ;YES
4353 3012     05000          JR      NC,NOSPCS       ;OUT OF RANGE
4355 FE0D     05050          CP      GTOTOK-80H      ;IS COMMAND 'GOTO'
4357 300E     05100          JR      C,NOSPCS        ;OUT OF RANGE
4359 2038     05150          JR      Z,CGOTO         ;COMMAND IS GOTO
435B FE11     05200          CP      GSBTOK-80H      ;IS COMMAND 'GOSUB'?
435D 2826     05250          JR      Z,CGOSUB        ;YES
435F FE1F     05300          CP      RSNTOK-80N      ;IS CONMAND 'RESUME'
4361 2007     05350          JR      Z,CRSN          ;YES
4363 FE0F     05400          CP      IFTOK-80N       ;IS COMMAND 'IF'?
4365 2073     05450          JR      Z,CIF           ;YES
              05500 ;END SPECIAL CASE CHECK
4367 C36A1D   05550 NOSPCS   JP      1D6AH           ;NOT SPECIAL CASE
              05600
              05650 ;COMMAND RESUME
436A 11F240   05700 CRSN     LD      DE,40F2H        ;POINT DE TO ERROR #
436D 1A       05750          LD      A,(DE)          ;ERROR # TO A
436E B7       05800          OR      A               ;NO ERROR?
436F CAA819   05850          JP      Z,19A8H         ;RESUME CALL ILLEGAL
4372 3C       05900          INC     A
4373 329A40   05950          LD      (409AH),A
4376 12       06000          LD      (DE),A
4377 D7       06050          RST     10H             ;NEXT BYTE TO A
4378 DAC11F   06100          JP      C,1FC1H         ;NUMBER
437B CAC11F   06150          JP      Z,1FC1H         ;END STATEMENT
437E FE87     06200          CP      87H             ;NEXT?
4380 CACD1F   06250          JP      Z,1FCDH         ;YES
4383 1815     06300          JR      LNLBL           ;PROCESS AS LABEL
              06350
              06400 ;COMMAND GOSUB. SEE CGOTO
4385 0E03     06450 CGOSUB   LD      C,03            ;3 ADDITIONS TO STACK
4387 CD6319   06500          CALL    1963H           ;ROOM IN RAM?
438A C1       06550          POP     BC              ;SAVE RETURN
4388 E5       06600          PUSH    HL              ;SAVE CURR.LOCATION
438C ED5BA240 06650          LD      DE,(LNNBUF)     ;LINE # TO DE
4390 D5       06700          PUSH    DE              ;SAVE LINE #
4391 3E91     06750          LD      A,91H           ;GOSUB TOKEN
4393 F5       06800          PUSH    AF              ;  TO STACK
4394 33       06850          INC     SP              ;F REG.NOT NEEDED
4395 C5       06900          PUSH    BC              ;RESTORE RETURN
              06950
              07000 ;COMMAND GOTO: JUMP TO LABEL SEARCH ROUTINE IS REQUIRED, ELSE BASIC
              07050 ;ENTRY: HL POINT TO COMMAND TOKEN IN BASIC LINE
              07100 ;EXIT: HL POINT TO FIRST CHARACTER OF SOURCE LABEL OR LINENUMBER
              07150 ;       (NOT APPLICABLE FOR ALL CASES OF RESUME)
```

Program continues

E · X · P · A · N · D

YOUR TRS-80* or S-100



INTRODUCING THE XTD-TRS INTERFACE CARD FOR THE STD BUS

This card permits direct connection between the TRS-80* and the STD BUS system. The TRS-80* can even be used as a development system for Z-80 STD BUS. QC MicroSystems distributes a full line of STD BUS products from a number of manufacturers including Mostek, Xitex, Intelligence Systems, Advanced Technology, Antona Corp & others.

200 NS MEMORY!!

High speed 4116 RAMS for Maximum Reliability from your TRS-80*

SET OF 8 FOR $44

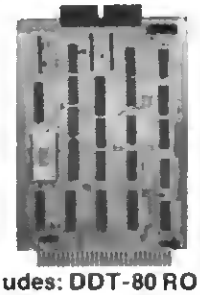
OTHER RAMS (MOSTEK)	$(1-9)
2114 UCB 1K X 4 450 NS	5.00
4118N-4 1K x 8 250 NS	24.00
4104N-4 4K x 1 250 NS	10.50
PROMS (MOSTEK)	
2716T/J12 2K x 8 650 NS	13.75
2716T/J8 2K x 8 450 NS	14.50

Z80 PARTS	2.5 MHZ $(1-9)	4.0 MHZ $(1-9)
Z80-CPU	9.50	13.50
Z80-PIO	6.00	9.50
Z80-CTC	6.00	9.50
Z80-DMA	20.00	29.00
Z80-SIO	17.75	23.50

MISC.
12" Video Monitor B&W $149.00

Xitex XTD-TRS Interface Card

DDT-80 ROM



$260 Includes: DDT-80 ROM

OFF THE SHELF STD BUS PRODUCTS

AVAILABLE NOW:

MDX-CPU1	$260	Z80 CPU/RAM/PROM
MDX-CPU2	$295	Z80 CPU/RAM/PROM
MDX-ORAM 8/32	*	Dynamic RAM
MDX-PIO	$250	Parallel I/O
MDX-A/D 8,10,12	*	A/D Converters
MDX-D/A 8, 12	*	D/A Converters
SYS-CPM*	*	CP/M 2.2 Disk S.W.
MDX-MATH	$699	Floating Point Math
MDX-SIO	$260	Serial I/O
XTD-VOT	*	Video Interface
PROM-I	$165	PROM Programmer
MDX-EPROM/UART	$225	Combination PROM/UART
CARD CAGES	*	8-22 Slot with MotherBoard
POWER SUPPLY	$135	

*Contact QC for Pricing Options

Ask For Our Full Catalog Of Products And Services

MasterCharge, Visa, American Express, Check, C.O.D. accepted. Add $3.00 for U.S. Shipping & Handling. Allow two weeks for shipment. Min. order of $10.00. All products covered by a 90 day OEM warranty. Prices subject to change without notice.

Oversees Inquiries Welcome!

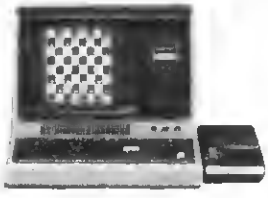
QC Micro Systems ✓395

P.O. BOX 401326
GARLAND, TEXAS 75040
(214) 343-1282

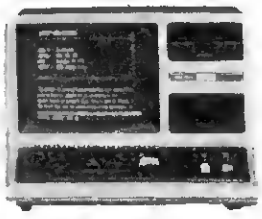
Ask for our complete catalogue!

*TRS-80 and CP/M are registered trademarks of Tandy Corp. and Digital Research, respectively

AUTHORIZED TRS 80® DEALER #R491



26 – 1056
16K Level II with Keypad
$670.00



26 – 1062
Model III 16K RAM
Model III, BASIC
$875.00

26 – 4002
Model II, 64K
$3500.00

WE ACCEPT CHECK, MONEY ORDER, OR PHONE ORDERS WITH VISA OR MASTER CHARGE. SHIPPING COSTS WILL BE ADDED TO CHARGE ORDERS. DISK DRIVES, PRINTERS, PERIPHERALS, AND SOFTWARE – YOU NAME, WE'VE GOT IT. WRITE OR CALL FOR OUR COMPLETE PRICE LIST.

FULL FACTORY WARRANTY ON ALL ITEMS SOLD.

C & S ELECTRONICS, LTD. 32 EAST MAIN ST. MILAN, MICH. 48160
✓145
(313) 439-1508 (313) 439-1400

C & S ELECTRONICS MART IS AN AUTHORIZED TRS 80® SALES CENTER STORE #R491

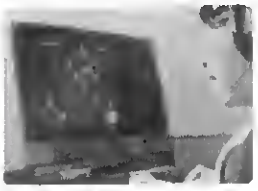
```
4396 D7       07200 CGOTO   RST     10H             ;NEXT BYTE
4397 DAC21E   07250         JP      C,1EC2H         ;NUMBER, JP TO ROM
              07300
              07350 ;LINE LABEL: SEARCHES BASIC PROGRAM FOR A LINE BEGINNING WITH LABEL IDENTIFIER '*'.
              07400 ;       CHECKS FOR SOURCE AND OBJECT STRING MATCH, UL ERROR AND CN ERROR
              07450 ;ENTRY: HL POINT TO FIRST CHARACTER OF SOURCE LABEL
              07500 ;EXIT: HL POINTS TO BYTE BEFORE OBJECT ROUTINE (END OF LINE WITH OBJ LABEL).
439A 2B       07550 LNLBL   DEC     HL
439B CD6144   07600         CALL    NXTBYT          ;NEXT BYTE TO A
439E CA9719   07650         JP      Z,SNERR         ;END STATEMENT,ERROR
43A1 0ECF     07700         LD      C,LBIDNT        ;LABEL IDENT. TO C
43A3 226B46   07750         LD      (SRLBPT),HL     ;SAVE ADD.SOURCE LABEL
43A6 CD2044   07800         CALL    CFKMS           ;FIND OBJ.LABEL
43A9 D2D91E   07850         JP      NC,ULERR        ;NO MATCH FOUND
43AC EB       07900         EX      DE,HL
43AD 7E       07950         LD      A,(HL)          ;END
43AE 23       08000         INC     HL              ;  OF
43AF B6       08050         OR      (HL)            ;    PROGRAM?
43B0 1E20     08100         LD      E,32D           ;CN ERROR
43B2 CAA219   08150         JP      Z,ERROR         ;YES, ERROR
43B5 2B       08200         DEC     HL              ;POINT TO
43B6 2B       08250         DEC     HL              ;  END PREV.LINE
43B7 C9       08300         RET
              08350
              08400 ;CONMAND ON: SEE CGOTO
43B8 D7       08450 CON     RST     10H             ;NEXT BYTE TO A
43B9 FE9E     08500         CP      9EN             ;ERROR TOKEN?
43BB 204D     08550         JR      Z,CONER         ;YES
43BD CD1C2B   08600         CALL    2B1CH           ;EVALUATE ARGUENENT
43C0 7E       08650         LD      A,(HL)          ;NEXT BYTE TO A
43C1 47       08700         LD      B,A             ;SAVE IN B
43C2 FE9I     08750         CP      91B             ;GOSUB?
43C4 2003     08000         JR      Z,CON1          ;YES
43C6 CF       08050         RST     08N
43C7 8D       08900         ADC     A,L
43C0 2B       08950         DEC     BL
43C9 4B       09000 CON1    LD      C,E             ;ARGUEMENT TO C
43CA 0D       09050 CNTCMA  DEC     C               ;SPECIFIED 0?
43CB 2004     09100         JR      NZ,NXTCMA       ;NO, NEXT COMMA
43CD 78       09150         LD      A,B             ;RESTORE A
43CE C34543   09200         JP      81D60           ;EVALUATE & DO
43D1 D7       09250 NXTCMA  RST     10N             ;NEXT BYTE TO A
43D2 B7       09300         OR      A               ;END STATEMENT?
43D3 C0       09350         RET     Z               ;YES
43D4 FE2C     09400         CP      ','             ;COMMA?
43D6 20F9     09450         JR      NZ,NXTCMA       ;NO
43D0 10F0     09500         JR      CNTCMA          ;YES, COUNT IT
              09550
              09600 ;CONMAND IF: SAME AS BASIC
              09650 ;ENTRY: HL POINTS TO IF TOKEN
              09700 ;EXIT: HL POINTS TO ARGUEMENT  OR   END OF LINE
43DA D7       09750 CIF     RST     10H             ;NEXT BYTE TO A
43DB CD3723   09000         CALL    2337H           ;EVALUATE ARGUEMENT
43DE 7E       09050         LD      A,(HL)          ;NEXT BYTE TO A
43DF FE2C     09900         CP      ','             ;COMMA?
43E1 CC701D   09950         CALL    Z,1D70H         ;YES, NEXT BYTE
43E4 FECA     10000         CP      0CAH            ;THEN?
43E6 CC701D   10050         CALL    Z,1D70R         ;YES, NEXT BYTE
43E9 2B       10100         DEC     BL
43EA E5       10150         PUSH    BL
43EB CD9409   10200         CALL    09948
43EE E1       10250         POP     BL
43EF 2007     10300         JR      Z,B2056
43F1 D7       10350 B204F   RST     10B
43F2 DAC2IE   10400         JP      C,1EC2H
43F5 C33F43   10450         JP      B1D5F           ;JUMP TO 'PATCH'
43F0 1601     10500 B2056   LD      D,01
43FA CD051F   10550 B2050   CALL    1F05N
43FD B7       10600         OR      A
43FE C8       10650         RET     Z
43FF D7       10700         RST     10H
4400 FE95     10750         CP      95H
4402 20F6     10000         JR      NZ,B2050
4404 15       10050         DEC     D
4405 20F3     10900         JR      NZ,B2050
4407 15       10950         DEC     D
4400 10E7     11000         JR      B204F
              11050
              11100 ;CONMAND ON ERROR: SEE CGOTO
440A D7       11150 CONER   RST     10N
4408 CF       11200         RST     08H
440C 8D       11250         ADC     A,L
440D 2B       11300         DEC     NL
440E D7       11350         RST     10H             ;NEXT BYTE
440F DA731F   11400         JP      C,1F73H         ;NUMBER, NORMAL BASIC
4412 E5       11450         PUSH    HL
4413 D7       11500 CONER1  RST     10H             ;NEXT BYTE,END STATEMENT?
4414 20FD     11550         JR      NZ,CONER1       ;NO
4416 E3       11600         EX      (SP),HL         ;YES, SAVE IN STACK
```

Program continues

```
4417 CD9A43   11650          CALL    LNLBL           ;FIND OBJECT LINE
441A 23       11700          INC     HL              ;POINT TO BEG.OBJ.LINE
441B 22F040   11750          LD      (40F0H),HL      ;SAVE ADD.LINE
441E E1       11800          POP     HL
441F C9       11850          RET
              11900
4420 ED5BA440 11950 CFKMS    LD      DE,(BPRPTR)     ;POINT DE TO BASIC PROG.
4424 CD3344   12000 CFKMS1   CALL    FNDKEY          ;FIND LINE BEG.WITH KEY
4427 B7       12050          OR      A               ;IS A 0?
4428 C8       12100          RET     Z               ;YES, END PROGRAM
4429 D7       12150 CFKMS2   RST     10H             ;NEXT BYTE TO A
442A EB       12200          EX      DE,HL           ;DE POINTS TO OBJ.LABEL
442B E5       12250          PUSH    HL              ;SAVE ADD. NEXT LINE
442C CD4444   12300          CALL    STRMCH          ;MATCH STRINGS
442F D1       12350          POP     DE              ;ADD.NEXT LINE
4430 30F2     12400          JR      NC,CFKMS1       ;NO MATCH, TRY AGAIN
4432 C9       12450          RET
              12500
              12550 ;FINDKEY: SEARCH BASIC PROGRAM FOR LINES BEGINNING WITH SEARCH KEY
              12600 ;ENTRY: SEARCH KEY IN REGISTER C
              12650 ;       DE POINTS TO LINE WHERE SEARCH IS TO BEGIN
              12700 ;EXIT: A=0 IF END OF PROGRAM; IX POINTS TO LINE WITH LABEL
              12750 ;       HL POINTS TO ADDRESS OF OBJECT KEY; DE POINTS TO THE NEXT LINE, THE
                    OBJECT ROUTINE
              12800 ;       C = SEARCH KEY
4433 EB       12850 FNDKEY   EX      DE,HL           ;START ADD. DE TO HL
4434 E5       12900          PUSH    HL              ;SAVE
4435 DDE1     12950          POP     IX              ;  IN IX
4437 7E       13000          LD      A,(HL)          ;END OF
4438 5F       13050          LD      E,A
4439 23       13100          INC     HL              ;  PROGRAM
443A B6       13150          OR      (HL)            ;     ?
443B C8       13200          RET     Z               ;YES
443C 56       13250          LD      D,(HL)          ;DE POINTS TO NEXT LINE
443D 23       13300          INC     HL              ;FIRST BYTE
443E 23       13350          INC     HL              ;  TEXT
443F D7       13400          RST     10H             ;     TO  A
4440 B9       13450          CP      C               ;=SEARCH KEY?
4441 20F0     13500          JR      NZ,FNDKEY       ;NO, TRY NEXT LINE
4443 C9       13550          RET
              13600
              13650 ;STRING MATCH: COMPARES TWO STRINGS, IMBEDDED SPACES IGNORED. DELIMITERS ARE
                    ':' ',' AND 0
              13700 ;ENTRY: ADDRESS OF SOURCE LABEL IN (SRLBPT); DE POINTS TO OBJECT LABEL
              13750 ;EXIT: CARRY SET IF MATCH FOUND
4444 2A6846   13800 STRMCH   LD      HL,(SRLBPT)     ;POINT TO SOURCE
4447 2B       13850          DEC     HL
4448 1B       13900          DEC     DE
4449 CD6144   13950 STRMC1   CALL    NXTBYT          ;NEXT BYTE TO A
444C 47       14000          LD      B,A             ;SAVE IN B
444D EB       14050          EX      DE,HL           ;POINT HL TO OBJ.WORD
444E 280A     14100          JR      Z,ENDSCE        ;JP IF END SOURCE WORD
4450 CD6144   14150          CALL    NXTBYT          ;NEXT BYTE OBJ.WORD
4453 EB       14200          EX      DE,HL
4454 C8       14250          RET     Z               ;MATCH FOUND
4455 B8       14300          CP      B               ;BYTES MATCH?
4456 20F1     14350          JR      Z,STRMC1        ;YES,COMPARE NEXT BYTES
4458 AF       14400 NOMCH    XOR     A               ;CANCEL CARRY, NO MATCH
4459 C9       14450          RET
445A CD6144   14500 ENDSCE   CALL    NXTBYT          ;ALSO END OBJ.WORD?
445D 20F9     14550          JR      NZ,NOMCH        ;NO, NO MATCH
445F 37       14600          SCF                     ;SET CARRY
4460 C9       14650          RET
              14700
4461 D7       14750 NXTBYT   RST     10H             ;RETURNS NEXT BYTE IN A
4462 C8       14800          RET     Z               ; AND Z FLAG SET
4463 FE2C     14850          CP      ','             ; IF ','  ':' OR 0
4465 C9       14900          RET
              14950
              15000 ;LENGTH STRING: COUNTS LENGTH STRING TO DELIMITER
              15050 ;ENTRY: HL POINTS TO STRING
              15100 ;EXIT: B = LENGTH OF STRING
4466 0600     15150 LENSTR   LD      B,00            ;INITIALIZE COUNTER
4468 7E       15200 LENST1   LD      A,(HL)          ;NEXT BYTE
4469 B7       15250          OR      A               ;0 ?
446A C8       15300          RET     Z               ;YES, END STRING
446B FE3A     15350          CP      ':'             ;':' ?
446D C0       15400          RET     Z               ;YES, END STRING
446E FE2C     15450          CP      ','             ;COMMA?
4470 C8       15500          RET     Z               ;YES, END STRING
4471 04       15550          INC     B               ;COUNT IT
4472 23       15600          INC     HL              ;POINT TO NEXT BYTE
4473 18F3     15650          JR      LENST1
              15700
              15750 ;COMMAND LABEL: CHECKS FOR BASIC LINE OR COMMAND ENTERED
              15800 ;ENTRY: USER ENTRY IN BASIC INPUT BUFFER
4475 CDC646   15850 CMDLBL   CALL    INTOFF          ;DISABLE INTERRUPTS ?
4478 F5       15900          PUSH    AF
```

Program continues

Games from BIG FIVE will turn your computer into a TRS-80 HOME ARCADE

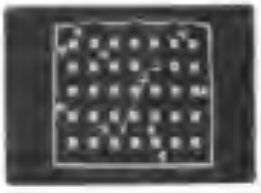


SUPER NOVA©

If you and your TRS-80 have longed for a fast-paced arcade-type game that is truly a challenge, then **SUPER NOVA** is what you've been waiting for. In this two player machine-language game, large asteroids float ominously around the screen. Suddenly your ship appears and you must destroy the asteroids before they destroy you! (But watch out because big asteroids break apart into little ones.) The controls that your ship will respond to are thrust, rotate, hyperspace, and fire. All right! You've done it! You've cleared away all the asteroids! But what is that saucer with the laser doing? Quick! You must destroy him fast because that guy's accurate!

GALAXY INVASION©

The sound of the klaxon is calling you! Cruel and crafty invaders have been spotted in battle formation warping toward Earth at an incredible speed. Suddenly, your ship materializes just below the huge flock of invaders. Quickly and skillfully you shift right and left as you carefully fire your lasers at them. But watch out! A few are breaking out of the convoy and flying straight at you! As the whine of their engines gets louder, you place your finger on the fire button knowing all too well that this shot must connect—or your mission will be permanently over! With sound effects!



ATTACK FORCE©

Your TRS-80 screen has been transformed into a maze-like playfield for this game. As your ship appears on the bottom of the screen, eight alien warships appear on the top. All of them are traveling at flank speed directly at you! Quickly and boldly you move toward them and fire missiles to destroy them. But the more aliens you destroy, the faster the remaining ones become. If you get too good you must endure the wrath of the keeper of the mazefield: the menacing "Flagship". You must destroy him fast because, as you will find out, that guy's accurate! With sound effects!

COSMIC FIGHTER©

With thousands of stars whizzing by you, your **SPACE DESTROYER** ship comes out of hyperspace directly under a convoy of aliens. Almost effortlessly, you skillfully destroy every last one. But before you can congratulate yourself, another set appears. These seem to be slightly more intelligent than the first set. Quickly you eliminate all of them, too. But your fuel supply is rapidly diminishing. You must still destroy two more sets before you can dock with your space station. All right! The space station is now on your scanners! Oh no! Intruders have overtaken the station! You must skillfully fire your neutron lasers to eliminate the intruders from the station before your engines run out of fuel and explode! With sound!

METEOR MISSION II©

The second **Big Bang** has occurred and the galaxy is full of stray asteroids and meteors. As you look through your space port you see a belt of asteroids drifting across the screen blocking your path to the safety of the space station above. But be careful because meteor showers, exploding suns and invading aliens may strike your ship and send it hurtling back to ground level. How many times can you and your opponent maneuver through those obstacles before time runs out? With sound effects!

BIG FIVE SOFTWARE

P.O. Box 9078-185, Van Nuys, California 91409

✓357

Prices per game. Level I—$14.95, Level II—$14.95, Level II/Disk—$17.95
Specify which version when ordering.
10% discount for 2 games, 15% for 3 or more.
Please add $1.00 postage/handling, Calif. residents add 8% tax.
All games are written in machine language and supplied on cassette.
Disk versions save high scores to your TRSDOS or NEWDOS diskette.

Cassette versions require 16K memory, disk versions require 32K.
Write for info. on Mod 3 versions.
All games ©1980 by Bill Hogue & Jeff Konyu.
TRS-80 & TRSDOS are trademarks of Tandy Corp.
NEWDOS is a trademark of Apparat, Inc.
Dealer inquiries invited.

Give Card Number, Expiration Date and Signature for Master Charge and VISA orders.

✓Reader Service—see page 274

```
4479 08          15950          EX      AF,AF'          ;SAVE AF
447A D9          16000          EXX                     ;SAVE REGISTERS
447B F1          16050          POP     AF              ;RESTORE AF
447C 3837        16100          JR      C,LINEIN        ;BASIC LINE ENTERED
                 16150 ; LABEL MENU: CHECKS INPUT AGAINST LABEL COMMANDS
                 16200 ;EXIT: C = NUMBER OF COMMAND, 0 FOR NO MATCH
447E 0E00        16250 LBLMNU   LD      C,00            ;INITIALIZE COUNTER
4480 11AD45      16300          LD      DE,WRDLST       ;POINT TO START OF LIST
4483 2AE64B      16350 LBLMN1   LD      HL,(40E6H)      ;POINT TO ENTRY IN BUFF.
4486 D7          16400          RST     10H             ;FIRST BYTE OF ENTRY
4487 BC          16450          INC     C               ;INCREASE COUNTER
4488 1A          16500 NXTWRD   LD      A,(DE)          ; BYTE TO A
4489 B7          16550          OR      A               ;END WORD?
448A 13          16600          INC     DE              ;POINT TO NEXT BYTE
448B 28FB        16650          JR      NZ,NXTWRD       ;NOT BEG.OF WORD
448D 1A          16700 LBLMN2   LD      A,(DE)          ;FIRST BYTE WORD
448E FE00        16750          CP      00H             ;END LIST?
4490 2002        16800          JR      NZ,NXTWR1       ;NO
4492 AF          16850          XOR     A               ;YES
4493 4F          16900          LD      C,A             ;A IS 0 FOR NO MATCH
4494 B7          16950 NXTWR1   OR      A               ;END WORD?
4495 2007        17000          JR      Z,JMPR          ;YES, FIND ROUTINE
4497 BE          17050          CP      (HL)            ;NO, BYTES = ?
4498 20E9        17100          JR      NZ,LBLMN1       ;NO MATCH, TRY NEXT WORD
449A D7          17150          RST     10H             ;NEXT BYTE
449B 13          17200          INC     DE              ;
449C 1BEF        17250          JR      LBLMN2          ;BYTES MATCH SO FAR
                 17300 ;JUMPER: JUMPS TO ROUTINE CORRESPONDING TO COMAND
                 17350 ;ENTRY: REG. C = NUMBER OF WORD IN WORD LIST MATCHING ENTRY; 0 IF NO MATCH
449E 0600        17400 JMPR     LD      B,00            ;
44A0 CB21        17450          SLA     C               ;NUMBER * 2
44A2 E5          17500          PUSH    HL              ;
44A3 21B644      17550          LD      HL,JMPTBL       ;POINT TO JUMP TABLE
44A6 09          17600          ADD     HL,BC           ;ADDRESS
44A7 7E          17650          LD      A,(HL)          ;  OF
44A8 23          17700          INC     HL              ;    ROUTINE
44A9 66          17750          LD      H,(HL)          ;     TO
44AA 6F          17800          LD      L,A             ;       HL
44AB E3          17850          EX      (SP),HL         ;ADDRESS TO STACK
44AC C9          17900          RET                     ;JUMP TO ROUTINE
44AD D9          17950 NOMENU   EXX                     ;NO MATCH,RESTORE REG'S
44AE 08          18000          EX      AF,AF'          ;RESTORE AF
44AF CDBC46      18050          CALL    INTON           ;ENABLE INTERUPTS?
44B2 C37544      18100 EXCMDL   JP      CMDLBL          ;EXCHANGED WITH BASIC RAM
                 18150
                 18200 ;BASIC LINE INPUT: LB ERROR CHECK
                 18250 ;ENTRY: BASIC LINE IN INPUT BUFFER. LINE NUMBER IN DE
                 18300 ;       IF MATCH IS FOUND AND LINE NUMBER ARE THE SAME, LINE IS BEING EDIYED.
44B5 2AE640      18350 LINEIN   LD      HL,(40E6H)      ;POINT TO ENTRY IN BUFF
44B8 D7          18400          RST     10H             ;FIRST BYTE ENTRY
44B9 BECF        18450          LD      C,LBIDNT        ;IS IT
44BB B9          18500          CP      C               ;  LABEL IDENTIFIER?
44BC 20EF        18550          JR      NZ,NOMENU       ;NO, IGNORE
44BE CD6144      18600          CALL    NXTBYT          ;ENTRY?
44C1 2BEA        18650          JR      Z,NOMENU        ;NO
44C3 22A945      18700          LD      (ERLBPT),HL     ;SAVE ADDRESS LABEL
44C6 CD2044      18750          CALL    CFKMS           ;MATCH IN PROGRAM?
44C9 3BE2        18800          JR      NC,NOMENU       ;NO, OK
44CB D9          18850          EXX                     ;
44CC D5          18900          PUSH    DE              ;GET LINE # ENTRY
44CD D9          18950          EXX
44CE D1          19000          POP     DE
44CF DD6EB2      19050          LD      L,(IX+2)        ;LINE # OBJECT
44D2 DD6603      19100          LD      H,(IX+3)        ;  TO HL
44D5 DF          19150          RST     18H             ;SAME #?
44D6 20D5        19200          JR      Z,NOMENU        ;YES, SAME LINE (EDITED)
44D8 C39246      19250          JP      LBLERR          ;LABEL ALREADY ASSIGNED
                 19300
44DB CDE444      19350 ENTLBL   CALL    LBLBAS          ;ENTER LABEL TO BASIC BLOCK
44DE CD5D1B      19400 EXTLBL   CALL    BASINI          ;INIYIALIZE FOR BASIC
44E1 C3B646      19450          JP      READY           ;JUMP TO READY
                 19500
44E4 7E          19550 LBLBAS   LD      A,(HL)          ;ENTER ON CMD 'BASIC..'
44E5 B7          19600          OR      A               ;0 ?
44E6 2B5B        19650          JR      Z,CHECK         ;YES, NO SWITCH
44E8 FE93        19700          CP      REMTOK          ;REM TOKEN?
44EA 280E        19750          JR      Z,REMLBL        ;YES
44EC FEB6        19800          CP      DELTOK          ;DELETE TOKEN?
44EE 2B1C        19850          JR      Z,DELLBL        ;YES
44F0 1BCB        19900          JR      EXCMDL          ;NOT RECOGNIZED
                 19950
                 20000 ;DELETE REMARKS
44F2 0E93        20050 DELREM   LD      C,REMTOK        ;REM TOKEN IS SEARCH KEY
44F4 CD1145      20100          CALL    DELLB9          ;FIND AND DELETE
44F7 C3B546      20150          JP      READY
                 20200
                 20250 ;CONVERT TO LINE NUMBERS AND WRITE REMARKS FOR LABELS
44FA CD3845      20300 REMLBL   CALL    CHECK           ;LABEL TO BASIC
44FD 0ECF        20350          LD      C,LBIDNT        ;LABEL IDNT.IS SEARCH KEY
```

Program continues

Poor Man's Floppy

HIGH SPEED CASSETTE SYSTEM

FOR TRS-80*

Now the widely acclaimed JPC Cassette System is available for your TRS-80* computer. The price is only $90.00

TC-8 Cassette System
JPC Products
Albuquerque, NM
Kit: $90
Assembled: $120

by **Carl A. Kollar**

[Reprint of June 1980 Review, *80 Microcomputing*]

I guess I don't have to tell any TRS-80 owners how frustrating the cassette system that comes with the computer can be. Even with the factory mod that's available, the annoyance of loading and checking programs becomes just barely tolerable.

If you're like me, after you've just plunked down a chunk of money for a Level II 16K machine, "you ain't got nuttin left" for even one disk drive at 500 bucks apiece. So you suffer.

A reasonable alternative is the Exatron Stringy Floppy (ESF). This will cost you about 250 bucks and totally eliminates your loading and saving problems, automatically and fast. I've had one of these for about six months and love it!

But, if the price is still too steep, have I got a device for you!

The Device

The February 1980 issue of *Microcomputing* had an ad that intrigued the hell out of me. It was a high-speed cassette system by JPC Products acclaimed as a "poor man's floppy." It made all sorts of seemingly ridiculous claims such as "loads five times faster," "stores 50,000 bytes on a 10-minute cassette," "less than one bad load in a million bytes with the volume control anywhere between one and eight."

All this for a measly [90] bucks? How could this be? A call to Albuquerque answered a few questions: Yes, it had its own power supply, and, it stored programs five times faster because it utilized higher density data. The computer outputs the information at a higher rate out of the rear keyboard connector.

The ad had even claimed anyone could build it even if you have never soldered before. JPC would make it work, if you couldn't—for free. I was sold. I placed my order, and it arrived about two months later (parts shortage).

I work in electronics, so I found the unit exceptionally easy to build. It took about an hour. The manual is superb. (That's better than great.) It was clear, concise and exact with no ambiguities. Important parts placements are stressed (polarity markings on electrolytics, bands on diodes, etc.).

JPC was right! With these instructions, you couldn't go wrong. The board quality is excellent. It is double-sided and parts locations are clearly marked on the component side of the board. There are no jumper wires to install. JPC utilizes PC traces and plated-through holes for connections to traces on the other side of the board.

Also, there are absolutely no adjustments or settings to bother with.

The documentation is a sheaf of 8½ x 11 papers stapled together. It is written in the nicest format I've seen in a while. Each command and/or subjects is covered on its own sheet in large type. All explanations are in easy to read English—not computerese.

Commands and Features

SAVE"filename": Saves your BASIC program on cassette.
LOAD: Reads the next BASIC program from the cassette.
LOAD"filename": Searches for and loads the specified file from cassette.
LOAD? and LOAD?"filename": Reads file from cassette, and compares contents to memory.
LOADN: Prints a list of all the programs on a cassette, until interrupted by the "break" key.
LOADN"filename": Same as above except the tape will stop at the end of the program named.
KILL: Removes the file manager program from memory so that the extra memory can be used by large programs.
RSET: Allows the operator to rewind and position the tape on tape recorders that have these functions tied to the motor control jack.
RUN"filename": TC-8 searches for a specified program and runs it immediately.
PUT"filename": Same as SAVE "filename", except it is for use with system tapes.
GET: Same as LOAD, except it is for use with system tapes.
GET"filename": Same as LOAD "filename", except it is for use with system tapes.
GET? and GET?"filename": Same as LOAD? and LOAD?"filename", except it is for use with system tapes.
GETN and GETN"filename": Same as LOADN and LOADN"filename", except it is for use with system tapes.
OPEN: Required before cassette input or output of a data file can be attempted.
CLOSE: Required to end a cassette data file.
PRINT#: Allows numerical or string data to be output to a cassette file.
INPUT#: Allows numerical or string data to be input from a cassette file.

I haven't counted them, so I don't know about the "one load in a million bytes" claim, but my son, Anthony (age 11), loaded about 30 of his programs from his Radio Shack format tape to a new TC-8 format tape. He's run them all and found no bad loads.

Unlike the standard tape system, you can position your tape anywhere before the program you want and not have to look for a blank spot between programs. The TC-8 patiently waits for the program you want and then starts loading without getting confused by the portion of the previous program you just fed it.

Try that on your regular cassette system; you'll wear out the reset button. ■

ORDER NOW

To order your TC-8 kit, send your check or money order for $90.00 plus $3.50 postage and handling to JPC PRODUCTS CO., 12021 Paisano Ct., Albuquerque, NM 87112 (New Mexico residents add 4% sales tax). Credit card orders accepted by phone or mail. Personal checks will delay shipment. We will otherwise immediately ship you the TC-8 kit, the cabinet, the ribbon cable, the power adapter, an instruction manual, and a cassette containing the software.

```
44FF ED5BA440 20400         LD      DE,(BPRPTR)      ;POINT TO BASIC PROG.
4503 CD3344   20450 RENLB1  CALL    FNDKEY           ;FIND LINE WITH KEY
4506 B7       20500         OR      A                ;END PROG?
4507 C8       20550         RET     Z                ;YES
4508 3693     20600         LD      (HL),RENTOK      ;REPLACE WITH REN
450A 18F7     20650         JR      RENLB1           ;NEXT LINE
              20700
              20750 ;CONVERT TO LINE NUMBERS, DELETE LABELS
450C CD3045   20800 DELLBL  CALL    CHECK            ;LABEL TO BASIC
450F 8ECF     20850         LD      C,LBIDNT         ;LABEL IDNT.IS SEARCH KEY
4511 ED5BA440 20900 DELLB0  LD      DE,(BPRPTR)      ;POINT TO BASIC PROGRAM
4515 CD3344   20950 DELLB1  CALL    FNDKEY           ;FIND LINE BEG. WITH KEY
4518 B7       21000         OR      A                ;END PROG?
4519 C8       21050         RET     Z                ;YES
451A C5       21100         PUSH    BC               ;SAVE KEY
451B DDE5     21150         PUSH    IX               ;BEGINNING LINE
451D 2AF940   21200         LD      HL,(SCLRPT)      ;POINT TO VARIABLES
4520 ED52     21250         SBC     HL,DE            ;NUMBER OF BYTES TO MOVE
4522 E5       21300         PUSH    HL               ;  TO
4523 C1       21350         POP     BC               ;    BC
4524 E1       21400         POP     HL               ;BEG.LINE TO HL
4525 EB       21450         EX      DE,HL            ;
4526 EDB0     21500         LDIR                     ;MOVE & DELETE
4528 ED53F940 21550         LD      (SCLRPT),DE      ;NEW SCALAR POINTER
452C DDE5     21600         PUSH    IX               ;BEG. LINE
452E D1       21650         POP     DE               ;  TO DE
452F CDFC1A   21700         CALL    1AFCH            ;WRITE LINE POINTERS
4532 DDE5     21750         PUSH    IX               ;BEG. LINE
4534 D1       21800         POP     DE               ;  TO DE
4535 C1       21850         POP     BC               ;RESTORE SEARCH KEY
4536 18DD     21900         JR      DELLB1           ;NEXT LINE
              21950
              22000 ;CHECK: CHECKS BASIC PROGRAM FOR COMMAND WHICH MAY TAKE A LABEL
              22050 ;       CALLS APPROPRIATE PROCESSING ROUTINE & WRITES LINE NUMBER FOR LABEL
4538 2AA440   22100 CHECK   LD      HL,(BPRPTR)      ;POINT TO BASIC PROG.
453B 2B       22150         DEC     HL
453C 2B       22200         DEC     HL
453D 113D45   22250 CHECK0  LD      DE,CHECK0        ;RETURN ADDRESS
4540 D5       22300         PUSH    DE               ;  TO STACK
4541 D7       22350 CHECK1  RST     10H              ;NEXT BYTE
4542 2010     22400         JR      NZ,CHECK2        ;NOT END STATENENT
4544 FE3A     22450         CP      ':'              ;END STATENENT?
4546 28F9     22500         JR      Z,CHECK1         ;YES
4548 D1       22550         POP     DE               ;SAVE RETURN
4549 23       22600         INC     HL               ;ADDRESS LINE
454A E5       22650         PUSH    HL               ;  TO
454B FDE1     22700         POP     IY               ;    IY
454D 7E       22750         LD      A,(HL)           ;END
454E 23       22800         INC     HL               ;  OF
454F B6       22850         OR      (HL)             ;    PROG?
4550 C8       22900         RET     Z                ;YES
4551 D5       22950         PUSH    DE               ;RESTORE RETURN
4552 23       23000         INC     HL               ;LINE
4553 5E       23050         LD      E,(HL)           ;  NUMBER
4554 23       23100         INC     HL               ;    TO
4555 56       23150         LD      D,(HL)           ;       DE
4556 ED53A240 23200         LD      (LNMBUF),DE      ;SAVE LINE #
455A 18E5     23250         JR      CHECK1           ;BEGIN CHECK OF LINE
455C FEA1     23300 CHECK2  CP      ONTOK            ;ON TOKEN?
455E 2814     23350         JR      Z,PRCSON         ;YES, JUMP
4560 FE8D     23400         CP      GTOTOK           ;GOTO TOKEN?
4562 2020     23450         JR      Z,PRCSGG         ;YES, JUMP
4564 FE91     23500         CP      GSBTOK           ;GOSUB TOKEN?
4566 2024     23550         JR      Z,PRCSGG         ;YES, JUMP
4568 FE9F     23600         CP      RSMTOK           ;RESUME TOKEN?
456A 2801     23650         JR      Z,PRCSRM         ;YES, JUMP
456C C9       23700         RET
              23750
              23800 ;PROCESS RESUME: SEE PRCSGG
456D D7       23850 PRCSRM  RST     10H              ;NEXT BYTE
456E FE87     23900         CP      NXTTOK           ;RESUME NEXT?
4570 C8       23950         RET     Z                ;YES
4571 2B       24000         DEC     HL
4572 1818     24050         JR      PRCSGG           ;PROCESS AS  GOTO/GOSUB
              24100
              24150 ;PROCESS ON: SEE PRCSGG
4574 D7       24200 PRCSON  RST     10H              ;NEXT BYTE
4575 2002     24250         JR      NZ,PRCSO1        ;NOT END STATEMENT
4577 2B       24300         DEC     HL
4578 C9       24350         RET
4579 FE8D     24400 PRCSO1  CP      GTOTOK           ;ON GOTO?
457B 280A     24450         JR      Z,PRONOK         ;YES, OK
457D FE91     24500         CP      GSBTOK           ;ON GOSUB?
457F 2806     24550         JR      Z,PRONOK         ;YES, OK
4581 FE2C     24600         CP      ','              ;COMMA?
4583 2002     24650         JR      Z,PRONOK         ;YES, OK
4585 18ED     24700         JR      PRCSON           ;NEXT BYTE
4587 CD8C45   24750 PRONDK  CALL    PRCSGG           ;PROCESS AS GOTO/GOSUB
458A 18E8     24800         JR      PRCSON           ;NEXT BYTE
              24850
```

Program continues

```
                24900 ;PROCESS A GOTO OR GOSUB; CHANGES LABELS TO LINE NUMBERS
                24950 ;ENTRY: HL POINTS TO BYTE BEFORE FIRST CHARACTER OF LABEL OR LINE NUMBER
                25000 ;EXIT: HL POINTS TO NEXT BYTE TO BE CHECKED (BY CHECK) FORGOTO, GOSUB, ETC.
458C D7         25050 PRCSGG  RST     10H             ;NEXT BYTE
458D D8         25100         RET     C               ;NUMBER
458E 2002       25150         JR      NZ,PRCSG2       ;LABEL
4590 2B         25200 PRCSG1  DEC     HL
4591 C9         25250         RET
4592 FE2C       25300 PRCSG2  CP      ','             ;COMMA?
4594 28FA       25350         JR      Z,PRCSG1        ;YES
4596 CD9A43     25400         CALL    LNLBL           ;PROCESS A LABEL
4599 23         25450         INC     HL              ;LINE
459A 23         25500         INC     HL              ;  NUMBER
459B 23         25550         INC     HL              ;    OF
459C 5E         25600         LD      E,(HL)          ;       ROUTINE
459D 23         25650         INC     HL              ;         TO
459E 56         25700         LD      D,(HL)          ;           DE
459F CDAD45     25750         CALL    FNDLNE          ;'FOUND A LINE'
45A2 08         25800         EX      AF,AF'          ;RESTORE FLAG
45A3 D0         25850         RET     NC              ;LINE POINTERS OK
45A4 D9         25900         EXX                     ;SAVE REG'S
45A5 FDE5       25950         PUSH    IY              ;ADDRESS OF
45A7 D1         26000         POP     DE              ;  LINE TO DE
45A8 CDFC1A     26050         CALL    1AFCH           ;WRITE NEW LINE POINTERS
45AB D9         26100         EXX                     ;RESTORE REGISTERS
45AC C9         26150         RET
                26200
45AD CD0646     26250 FNDLNE  CALL    BINDEC          ;CONVERT TO DECIMAL
45B0 CDBA45     26300         CALL    SPACE           ;ADJ.SPACE IN PROG.
45B3 08         26350         EX      AF,AF'          ;SAVE FLAG
45B4 CDF545     26400         CALL    WRNUM           ;WRITE DECIMAL NUMBER
45B7 EB         26450         EX      DE,HL
45B8 2B         26500         DEC     HL              ;HL POINTS TO END LINE
45B9 C9         26550         RET
                26600
                26650 ;SPACE: MOVES PART OF BASIC PROGRAM IN RAM TO ALLOW A LINE NUMBER TO BE
                26700 ;       WRITTED IN PLACE OF A LABEL.
                26750 ;ENTRY: ADDRESS OF SOURCE LABEL IN (SRLBPT)
                26800 ;       LENGTH OF ASCII REPRESENTATION ON LINE NUMBER IN (LENDEC)
                26850 ;EXIT: CORRECT AMOUT OF 'SPACE' IN PROGRAM TO WRITE LINE NUMBER.
                26900 ;       CARRY SET IF NEW LINE POINTERS MUST BE WRITTEN
45BA 2A6B46     26950 SPACE   LD      HL,(SRLBPT)     ;POINT TO SOURCE LABEL
45BD CD6644     27000         CALL    LENSTR          ;COUNT STRING
45C0 3AD340     27050         LD      A,(LENDEC)      ;LENGTH ASCII REP.
45C3 B8         27100         CP      B               ;COMPARE TO LEN.STRING
45C4 C8         27150         RET     Z               ;SAME
45C5 08         27200         EX      AF,AF'          ;SAVE FLAG
45C6 EB         27250         EX      DE,HL           ;END STRING TO DE
45C7 48         27300         LD      C,B             ;COUNT TO
45C8 AF         27350         XOR     A               ;  BC &
45C9 47         27400         LD      B,A             ;   CANCEL CARRY
45CA 2AF940     27450         LD      HL,(SCLRPT)     ;POINT HL TO VARIABLES
45CD E5         27500         PUSH    HL              ;SAVE
45CE ED52       27550         SBC     HL,DE           ;NUMBER BYTES TO MOVE
45D0 E3         27600         EX      (SP),HL         ;TO STACK,RESTORE HL
45D1 08         27650         EX      AF,AF'          ;RESTORE FLAG
45D2 300F       27700         JR      NC,INCSP        ;NEED MORE SPACE IN PROG.
                27750 ;DECREASE SPACE IN BASIC PROGRAM
45D4 2A6B46     27800 DECSP   LD      HL,(SRLBPT)     ;POINT TO SOURCE LABEL
45D7 4F         27850         LD      C,A             ;LENGTH ASCII REP.TO BC
45D8 09         27900         ADD     HL,BC           ;DESTINATION TO DE
45D9 EB         27950         EX      DE,HL           ;  SOURCE TO HL
45DA C1         28000         POP     BC              ;BYTES TO MOVE TO BC
45DB EDB8       28050         LDIR                    ;MOVE & DELETE
45DD ED53F940   28100         LD      (SCLRPT),DE     ;NEW SCALAR POINTER
45E1 37         28150         SCF                     ;SET CARRY
45E2 C9         28200         RET
                28250 ;INCREASE SPACE IN BASIC PROGRAM
45E3 E5         28300 INCSP   PUSH    HL              ;SAVE VARIABLE POINTER
45E4 91         28350         SUB     C               ;NUMBER OF BYTES TO ADD
45E5 4F         28400         LD      C,A             ;  TO BC
45E6 09         28450         ADD     HL,BC           ;DESTINATION
45E7 CD6C19     28500         CALL    196CH           ;ENOUGH RAM?
45EA 22F940     28550         LD      (SCLRPT),HL     ;NEW SCALAR POINTER
45ED EB         28600         EX      DE,HL           ;DESTINATION TO DE
45EE E1         28650         POP     HL              ;SOURCE TO HL
45EF C1         28700         POP     BC              ;BYTES TO MOVE TO BC
45F0 03         28750         INC     BC              ;FUDGE FACTOR
45F1 EDB8       28800         LDDR                    ;MOVE, MAKE ROOM IN PROG
45F3 37         28850         SCF                     ;SET CARRY
45F4 C9         28900         RET
                28950
                29000 ;WRITE NUMBER: WRITE ASCII DECIMAL REPRESENTATION OF LINE NUMBER IN BASIC PROG.
                29050 ;ENTRY: NUMBER OF BYTES IN NUMBER IN (LENDEC)
                29100 ;       ASCI REP. OF NUMBER IN BUFFER POINTED TO BY (DECPTR)
                29150 ;       ADDRESS WHERE NUMBER TO BE WRITTEN POINTED TO BY (SRLBPT)
45F5 0600       29200 WRNUM   LD      B,00            ;INTIALIZE COUNTER
45F7 21D340     29250         LD      HL,LENDEC       ;POINT TO BUFFER
```

Program continues

FMG CORPORATION NOW CARRIES GRAHAM-DORIAN & PEACHTREE SOFTWARE

FMG CORPORATION
P.O. Box 16020
Fort Worth, Texas
76133
(817) 294-2510

✓12

M-530

All FMG Software Products Include All Necessary Manuals

NEW VERSATILITY
For Your TRS-80
CP/M®2

CONTROL PROGRAM FOR MICROCOMPUTERS ENABLING YOU TO RUN SOFTWARE PUBLISHED FOR CP/M 1.4 ON THE TRS-80

CP/M is considered the industry standard disk operating system because it gives you the hardware-independent interface you need to make your computer work for you. CP/M 2.0 is the latest in the evolution of a proven reliable and efficient software system. FMG CORPORATION NOW OFFERS THE CP/M 2.0 FOR THE TRS-80.
It features an enhanced upward compatible file system and powerful new random access capabilities. The CP/M 2.0 from FMG provides the ability to run software published for the CP/M system, on the TRS-80 Model II. From minidisks, floppy disks, all the way to high-capacity hard disks, the flexibility of CP/M 2.0 makes it a truly universal operating system. The package includes an 8" system disk, editor, assembler and debugger for the TRS-80

Available in Format A, B, C, G only . . . **$200/$25**

MP/M™
MULTI-PROGRAMMING MONITOR

NEW INDUSTRY STANDARD

A deluxe operating system that provides big computer facilities at small computer prices. MP/M is a monitor program which operates with your microcomputer to provide multi-terminal access with multi-programming at each terminal. Best of all, it's CP/M compatible which means you can run a wide variety or programming languages, applications packages, and development software.

You can run simultaneous editors, program translators, and background printer spoolers. Or you can use MP/M for data entry or data-base access from remote terminals. Or you can use MP/M real-time features to monitor an assembly line and automatically schedule programs for execution throughout the day. MP/M makes an excellent focal point for a cluster of connected microcomputers. The possibilities are limitless.

(Format B)	$450/$35
(Format G)	$300/$35

*CP/M and MP/M are *trademarks* of Digital Research.
Z80 is a trademark of Zilog, Inc.
TRS-80 is a trademark of Tandy Corp.
Pascal/M is a trademark of Sorcim.

· MICROPRO INTERNATIONAL

	SOFTWARE / WITH MANUAL	MANUAL ALONE

SUPER-SDRT I — Sort, merge, extract utility as absolutes executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point, exponential, field justified etc. Even variable number of fields per record! **$225/$25**

SUPER-SDRT II — Above available as absolute program only . . . . . **$175/$25**

SUPER-SDRT III — As II without SELECT/EXCLUDE . . . . . **$125/$25**

WORD-STAR — Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace. Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning . . . . . **$495/$40**

WORD-STAR Customization Notes — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR . . . . . **NA/$95**

WORD-MASTER Text Editor — In one mode has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal . . . **$150/$25**

FLOPPY SAVER — Protection for center holes of 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering peel, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.

5" Kit	$14.95
5" Rings only	$7.95
8" Kit	$16.95
8" Rings only	$8.95

HEAD CLEANING DISKETTE — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use

8"	$32.00
5¼"	$30.00

· DESPOOL — Allows flexibility and efficiency. (M) IDisk file printing can be accomplished while simultaneously using the computer for other tasks! Slower printers do not tie up the computer. Requires 32K minimum . . . . . **$75/$10**

· SCREEN EDIT — Text editor for program entry — allows user the ability to see entries as they are being made. Has command which enables user to move the viewed position of the file anywhere within the current dets file OR add information anywhere in the file. Requires 16K minimum . . . . . **$125/$25**
(Also available in TRS DOS format
Specify model or TRS-80)

· MAC — Disk-based, powerful macro assembler utilizes Standard Intel Mnemonics. Includes macro processor. (M)
The CP/M 8080 Macro Assembler reads assembly language statement from a diskette file and produces an Intel "HEX" format object file on the disk suitable for processing in the TRS-CP/M environment. Requires 32K minimum and CP/M . . . . . **$100/$25**

· ZSID — Efficient and reliable program testing system for Z80 microcomputers. Capabilities include traceback and histogram facilities. Allows real time break points. (M)
ZSID is a symbolic debugger which expands upon the features of the TRS-CP/M standard debugger, providing greatly enhanced facilities for assembly language program check-out. Requires 32K minimum and CP/M . . . . . **$99/$25**

MAIL LIST — Mailing list maintenance package. No sorting required to print normal address labels in zip code sequence. Supports new larger (M) zip code. Sorts and selects on multiple fields. Labels may be printed in user selectable formats. Includes sort and select utilities **$300/$35**

· PEACHTREE SOFTWARE SYSTEMS

GENERAL LEDGER — Records details of all financial transactions. Generates a balance sheet and an income statement. Flexible and adaptable design for both small businesses and firms performing client writeup services. Produces reports as follows: Trial Balance, Transaction Registers, Balance Sheet, Prior Year Comparative Balance Sheet, Income Statement, Prior Year Comparative Income Statement and Department Income Statements. Interactive with other PEACHTREE accounting packages. Supplied in source code for Microsoft BASIC . . . . . **$990/$30** (T)

ACCOUNTS PAYABLE — Tracks current and aged payables and incorporates a check writing feature. Maintains a complete vendor file with information on purchase orders and discount terms as well as active account status. Produces reports as follows: Open Voucher Report, Accounts Payable Aging Report and Cash Requirements. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . . **$990/$30** (T)

ACCOUNTS RECEIVABLE — Generates invoice register and complete monthly statements. Tracks current and aged receivables. Maintains customer file including credit information and account status. The current status of any customer account is instantly available. Produces reports as follows: Aged Accounts Receivable, Invoice Register, Payment and Adjustment Register and Customer Account Status Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . **$990/$30** (T)

PAYROLL — Prepares payroll for hourly, salaried and commissioned employees. Generates monthly, quarterly and annual returns. Prepares employee W-2's. Includes tables for federal withholding and FICA as well as withholding for all 50 states plus up to 20 cities from pre-computed or user generated tables. Will print checks, Payroll Register, Monthly Summary and Unemployment Tax Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC . . . . . **$990/$30** (T)

INVENTORY — Maintains detailed information on each inventory item including part number, description, unit of measure, vendor and reorder data, item activity and complete information on current item costs, pricing and sales. Produces reports as follows: Physical Inventory Worksheet, Inventory Price List, Departmental Summary Report, Inventory Status Report, The Reorder Report and the Period-to-Date and Year-to-Date reports. Supplied in source code for Microsoft BASIC . . . . . **$1,190/$30** (T)

MAILING ADDRESS — Keeps track of name and address information and allows the selective printing of this information in the form of mailing lists or address labels. Allows the user to tailor the system to his own particular requirements. User-defined format and print-out system uses a special format file which tells programs how to print the mailing list or address labels. Standard format files are included with system. Automatic sorting of data uses indexed file management routines which allow the name and address information to be sequentially retrieved and printed without file sorting. Supplied in source code for Microsoft BASIC . . . . . **$790/$90** (T)

· GRAHAM-DORIAN SOFTWARE SYSTEMS

GENERAL LEDGER — An on-line system; no batching is required. Entries to other GRAHAM-DORIAN (T) accounting packages are automatically posted. User satisfiables customized C.O.A. Provides transaction register, record of journal entries, trial balances and monthly closings. Keeps 14 month history and provides comparison of current year with previous year. Requires CBASIC-2 Supplied in source **$995/$35**

ACCOUNTS PAYABLE — Maintains vendor list and check register. Performs cash flow analysis. Flexible — writes checks to specific vendor for certain invoices or can make partial payments. Automatically posts to GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2 Supplied in source . . . . . **$495/$25** (T)

ACCOUNTS RECEIVABLE — Creates trial balance reports, prepares statements, ages accounts and records invoices. Provides complete information describing customer payment activity. Receipts can be posted to different ledger accounts. Entries automatically update GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source . . . . . **$955/$30** (T)

PAYROLL SYSTEM — Maintains employee master file. Computes payroll withholding for FICA, Federal and State taxes. Prints payroll register, checks, quarterly reports and W-2 forms. Can generate ad hoc reports and employee form letters with mail labels. Requires CBASIC-2 Supplied in source . . . . . **$399/$35** (T)

INVENTORY SYSTEM — Captures stock levels, costs, sources, sales, ages, turnover, markup, etc. Transaction information may be entered for reporting by salesman, type of sale, date of sale, etc. Reports available for accounting and decision making. Requires CBASIC-2 Supplied in source . . . **$395/$35** (T)

JOB COSTING — Designed for general contractors. To be used interactively with other GRAHAM-DORIAN accounting packages for tracking and analysing expenses. User establishes customized cost categories and job phases. Permits comparison of actual versus estimated costs. Automatically updates GRAHAM-DORIAN General Ledger or runs as stand alone system. Requires CBASIC-2 Supplied in source **$995/$35** (T)

Sample Program Disk For Each Graham-Dorian Business Package. Specify Package . . . **$45**

The sale of each proprietary software package conveys a license for use on one system only.

Prices F.O.B. Fort Worth. Tax, Shipping, handling and C.O.D. charges extra.

· MICROSOFT PRODUCTS

BASIC-80 — Disk Extended BASIC, ANSI compatible with long variable names, WHILE/WEND, chaining, variable length file records . . . . . **$350/$25** (M)

BASIC COMPILER — Language compatible with BASIC-80 and 3-10 times faster execution. Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAH-80 or COBOL-80 code modules . . . . . **$395/$25** (M)

FORTRAN-80 — ANSI 86 (except for COMPLEX) plus many extensions. Includes relocatable object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) . . . . . **$500/$25** (M)

COBOL-80 — Level 1 ANSI '74 standard COBOL plus most of Level 2. Full sequential, relative, and Indexed file support with variable file names, STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 . . . . . **$750/$25** (M)

MACRO-80 — 8080/Z80 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable linkable output. Loader, Library Manager and Cross References List utilities included **$150/$25** (M)

XMACRO-86 — 8086 cross assembler. All Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available . . . . . **$300/$25** (M)

PASCAL/M* — Compiler generates P code from extended language, implementation of standard PASCAL. Supports overlay structure through additional procedure calls and the SEGMENT procedure type. Provides convenient string handling capability with the added variable type STRING. Untyped files allow memory image I/O. Requires 56K CP/M **$150/$25** (M)

PASCAL/Z — Z80 native code PASCAL compiler. Produces optimized, RDMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, Microsoft Compatible relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M and Z80 CPU . . . . . **$395/$25** (M)

PASCAL/MT — Subset of standard PASCAL. Generates ROMable 8080 machine code. Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Version 3 includes Enumeration and Record data types. Manual explains BASIC to PASCAL conversion. Source for the run-time package requires Digital Research's MAC. Requires 32K . . . . . **$250/$30** (M)

CBASIC-2 Disk Extended BASIC — Non-interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining, integer and extended precision variables, etc. . . . **$110/$15** (M)

BSTAM — Utility to link one computer to another also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send a COM, etc. 9600 baud with wire, 300 baud with phone connection. Both ends need one. Standard and (M) versions can talk to one another. . . . . **$150/$5** (M)

SELECTOR III-C2 — Data Base Processor to create and maintain multi Key data bases. Prints formatted and sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments etc. Requires CBASIC-2 Supplied in source **$349/$20** (M) (T)

QLECTOR — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 52K system . . . . . **$250/$25** (T)

TEXTWRITER III — Text formatter to justify and paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* prepared files . . . . . **$125/$30** (M)

ORDERS MUST SPECIFY DISK SYSTEMS AND FORMATS:

FORMATS AVAILABLE:
(A) TRS-80 Model I (M) Keys Only
(B) TRS-80 Model II
(C) TRS-80 Model III (M) Keys Only
(D) HEATHKIT H89 (M) Keys Only
(E) NORTH STAR
(F) SUPER BRAIN QD
(G) STANDARD UNIMPLEMENTED

(M) Modified version available for use with CPM as implemented on Heath and TRS-80 Model I computers.

(T) For all (T) items listed above, the recommended system configuration consists of 48K CPM 2 full size disk drives, 24 x 80 CRT and 132 column printer.

FMG's LIBRARY:

PASCAL USER MANUAL & REPORT
(2nd) Edition by K. Jensen and N. Wirth

- **Tutorial Manual and Concise Reference Report for Both Programmers and Implementors**
- **Includes Helpful Examples to Demonstrate the Various Features of PASCAL**

The book consists of two parts: the user manual and the revised report. The manual is directed to those who have some familiarity with computer programming and who wish to get acquainted with the PASCAL language. The report defines standard PASCAL, which constitutes a common base between various implementations of the language.

Stock No.	Price
#521	$9.95

PASCAL PRIMER Problem Solving

This book has three major goals:
- **To introduce all aspects of the programming and problem solving process (includes problem specification and organization, algorithms, coding, debugging, testing, documentation and maintenance);**
- **To teach good programming style and how to produce a high quality finished product; and**
- **To teach the syntax of the PASCAL programming language**

Numerous examples are employed throughout the text. PASCAL is used as a vehicle to teach various aspects of programming techniques.

Stock No.	Price
#524	$18.95

Microcomputer Problem Solving Using Pascal
by Kenneth L. Bowles

- **A Book Designed for Both College Courses AND Individual Self-Study**
- **Ideal for use with UCSD Pascal**
- **Includes Extensions to Standard PASCAL**

This book is designed both for introductory courses in computer problem solving at the freshman and sophomore college level, and for individual self-study. It includes many examples and actually executable programs. It includes information on the necessary functions and procedures for handling graphics and strings.

Stock No.	Price
#522	$14.95

BEGINNER'S MANUAL FOR UCSD PASCAL SYSTEM

- **An Enlightening Introduction to UCSD PASCAL**
- **Demonstrates How to Use the UCSD PASCAL System and How to Program in PASCAL**
- **Includes Many Practical Examples of PASCAL Programs**

This book is intended to be used as an introduction and reference manual for people just beginning to use the UCSD Pascal Software System. Whether you have never used a computer before or whether you are an experienced programmer who is unfamiliar with UCSD PASCAL, this book will provide a relatively easy yet thorough introduction to UCSD PASCAL.

Stock No.	Price
#526	$11.95

PROGRAMING IN PASCAL
by Peter Grogono

- **An Excellent Introduction to One of the Fastest Growing Programming Languages Today**
- **Sections on Procedures and Files PLUS a Chapter on Dynamic Data Structures such as Trees and Linked Lists**

The text is arranged as a tutorial, containing both examples and exercises to increase reader proficiency in PASCAL. Concepts are illustrated by examples, ranging from the Tower of Hanoi problem to circumscribing a circle about a triangle. PROGRAMMING IN PASCAL is sure to hold the reader's interest.

Stock No.	Price
#523	$14.95

UCSD Reference Book

- **A Reference Guide to the Complete UCSD PASCAL System**
- **Includes information on Compiler Basic, Assembler and Editor**
- **Lists Actual P-Machine Codes**

This reference book can be a valuable and time-saving guide to through information on the UCSD PASCAL system. The easy-to-read manual provides fast access to pertinent data.

Stock No.	Price
#528	$25.00

✓Reader Service—see page 274

```
45FA 4E       29300          LD      C,(HL)          ;  ;LENGTH ASCII REP.TO C
45FB 23       29350          INC     HL              ;ADDRESS OF
45FC 5E       29400          LD      E,(HL)          ;  ASCII REP.
45FD 23       29450          INC     HL              ;    TO
45FE 56       29500          LD      D,(HL)          ;       DE
45FF 2A6B46   29550          LD      HL,(SRLBPT)     ;POINT TO SOURCE LABEL
4602 EB       29600          EX      DE,HL
4603 EDB0     29650          LDIR                    ;WRITE ASCII REP.OF #
4605 C9       29700          RET
              29750
              29800 ;BINARY TO DECIMAL: CONVERT INTEGER IN BINARY TO ASCII DECIMAL REPRESENTATION.
              29850 ;        SAVE ASCII REP. IN BUFFER, COUNT LENGTH
              29900 ;ENTRY: NUMBER IN DE
              29950 ;EXIT: ASCII REP. IN BUFFER POINTED TO BY (DECPTR), LENGTH IN (LENDEC)
4606 EB       30000 BINDEC   EX      DE,HL           ;ON ENTRY DE=#
4607 CD9A0A   30050          CALL    0A9AH
460A AF       30100          XOR     A
460B CD3410   30150          CALL    1034H
460E B6       30200          OR      (HL)
460F CDD90F   30250          CALL    0FD9H
4612 23       30300          INC     HL
4613 22D440   30350          LD      (DECPTR),HL     ;SAVE LOCATION ASCII REP.
4616 CD6644   30400          CALL    LENSTR          ;COUNT ASCII REP.
4619 78       30450          LD      A,B             ;SAVE
461A 32D340   30500          LD      (LENDEC),A      ;  LENGTH
461D C9       30550          RET
              30600
              30650 ;FINDS AND DISPLAYS LINE WITH LABEL IN BASIC PROGRAM.
              30700 ;CALED BY COMMAND FIND 'ROUTINE NAME'
461E CD9A43   30750 FINDER   CALL    LNLBL           ;FIND LABEL
4621 CD4646   30800          CALL    PRLNE           ;PRINT LINE WITH LABEL
4624 C3B646   30850          JP      READY
              30900
              30950 ;LLIST ALL LINES WITH LABEL IDENTIFIER
4627 3E01     31000 LLISTR   LD      A,01
4629 329C40   31050          LD      (409CH),A
              31100
              31150 ;LIST ALL LINES WITH LABEL IDENTIFIER
462C ED5BA440 31200 LISTER   LD      DE,(BPRPTR)     ;POINT TO BEG.BASIC PROG.
4630 0ECF     31250 LISTE1   LD      C,LBIDNT        ;LABEL IDNT.IS SEARCH KEY
4632 CD3344   31300          CALL    FNDKEY          ;FIND LINE
4635 B7       31350          OR      A               ;END PROG?
4636 CAB646   31400          JP      Z,READY         ;YES
4639 D5       31450          PUSH    DE              ;SAVE
463A DDE5     31500          PUSH    IX              ;BEG.LINE
463C E1       31550          POP     HL              ;  TO HL
463D CD4646   31600          CALL    PRLNE           ;PRINT LINE
4640 CD9B1D   31650          CALL    1D9BH           ;BREAK OR SHIFT@ ?
4643 D1       31700          POP     DE              ;RESTORE
4644 10EA     31750          JR      LISTE1          ;DO IT AGAIN
              31800
              31850 ;PRINT LINE: PRINTS A LINE OF THE BASIC PROGRAM
              31900 ;ENTRY: IX POINTS TO LINE
4646 DD6E02   31950 PRLNE    LD      L,(IX+2)        ;LINE #
4649 DD6603   32000          LD      H,(IX+3)        ;  TO HL
464C 22EC40   32050          LD      (40ECH),HL      ;ENABLES "." AS CURRENT LINE
464F DDE5     32100          PUSH    IX              ;ADDRESS OF LINE
4651 CDAF0F   32150          CALL    0FAFH           ;DISPLAY LINE #
4654 3E20     32200          LD      A,20H           ;SPACE
4656 CD2A03   32250          CALL    032AH           ;DISPLAY
4659 E1       32300          POP     HL              ;POINT TO LINE
465A 23       32350          INC     HL              ;POINT
465B 23       32400          INC     HL              ;  TO
465C 23       32450          INC     HL              ;    TEXT
465D 23       32500          INC     HL              ;      .
465E CD7E2B   32550          CALL    2B7EH           ;WRITE TO BUFFER
4661 2AA740   32600          LD      HL,(BUFPTR)     ;POINT TO BUFFER
4664 CD752B   32650          CALL    2B75H           ;DISPLAY
4667 CDFE20   32700          CALL    20FEH           ;ADVANCE CURSOR
466A C9       32750          RET
              32800
466B 0000     32850 SRLBPT   DEFW    0000H           ;SAVES ADD.SOURCE LABEL
              32900
              32950 ;WORD LIST: HOLD LIST OF WORDS RECOGNIZED BY LABEL.
              33000 ;LIST LOOKS WIERD BECAUSE OF IMBEDDED TOKENS (ROUT = R + OUT TOKEN, ETC.)
466D 00       33050 WRDLST   DEFB    00
466E 42       33100          DEFM    'BASIC'
4673 00       33150          DEFB    00
4674 B6       33200          DEFB    DELTOK
4675 9C       33250          DEFB    LNETOK
4676 93       33300          DEFB    REMTOK
4677 00       33350          DEFB    00
4678 B4       33400          DEFB    LSTTOR
4679 52       33450          DEFB    'R'
467A A0       33500          DEFB    OUTTOK
467B 00       33550          DEFB    00
467C 46       33600          DEFM    'FIND'
4680 00       33650          DEFB    0
4681 B5       33700          DEFB    LLSTOK
4682 52       33750          DEFB    'R'
```


Program continues

Fully TRS-80® Compatible

MPI B/51 DISK DRIVE

FREE CATALOG
Hundreds of other computer products at major savings. Cut out this coupon and send with your name and address.

$321.00

We pay UPS charges In Continental US

In Stock! Limited Quantities.

Includes: Case and Power Supply

Fully tested
Guaranteed for 90 days!

Here's why the MPI/B-51 is the drive for success:

- 40 tracks
- 5 ms track-to-track
- Auto-eject
- Hi-Temp stability
- Fully-closable door
- Speed constant <1½%
- Double density head
- Optical sensors—no switches
- 102K per disk

ADDS MORE POWER TO YOUR SYSTEM

Save time . . . Order by phone
Toll free: 1-800-323-4335
IN ILLINOIS CALL: 312-251-5955

TRS-80™ TANDY CORP

Other Money Savings Opportunities
Order by Phone or Mail

CABLES

2 Drive Cable	$29.00
4 Drive Cable	$39.00

Diskettes

Verbatim	10 for $31.00
BASF with Lib. Case	10 for $28.00
Dysan	5 for $21.00
Plastic File Box	$3.95

Operating Systems

TRSDOS 2.3	$14.95
40 Track Patch	$9.95
VTOS 4.0 "The Ultimate"	$99.00
NEWDOS + 40 track	$99.00
TRSDOS Manual	$5.95

TRS-80

16K Level II with keypad	$729.00
Expansion Interface ØK	$274.00

Printers

Centronics 779 (freight collect)	$849.00
Centronics 737-1	$815.00
IDS 460	$1219.00
Okidate Microline 80 with tracfors	$709.00
NEC Spinwriter 5530 (freight collect)	$2579.00

16K Memory Kits

Prime NEC 200ns dynamic RAM. Comes with complete instructions	$44.95

MPI Service Manual	$3.00
MPI Engineering Manual	$30.00

MIDWEST COMPUTER PERIPHERALS
P.O. BOX 437 • WILMETTE, ILLINOIS 60091

Midwest Computer Peripherals

P.O. Box 437 • Wilmette, Illinois 60091

✓8

ACCEPTED!

085

Quantity	Description	$ each	Total
		6% Ill. Tax	
		TOTAL	

☐ Check enclosed (Min Order $10.00)

Bill my ☐ Visa
☐ Master Charge
Acc. No. _____ Exp. _____

☐ Please send catalog

Name _____

Address _____

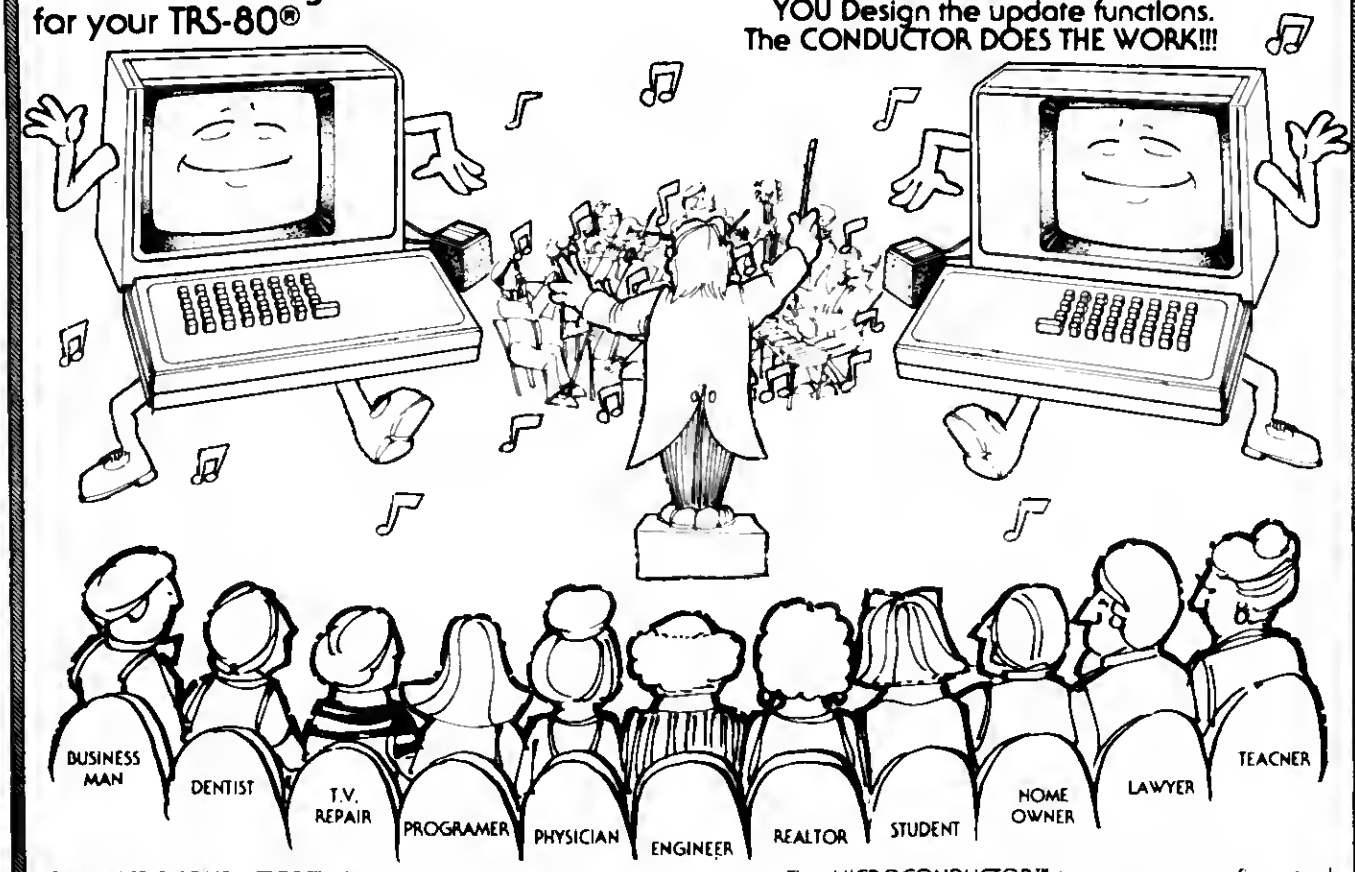
City _____ State _____ Zip _____

```
4603 A0        33000          DEFB    OUTTOR
4604 00        33050          DEFB    00
4605 00        33100          DEFB    00H
               33150
               34000 ;LOCATION OF ROUTINES CORRESPONDING TO WORD LIST
4606 AD44      34050 JMPTBL   DEFW    NOMENU
4608 0B44      34100          DEFW    ENTLBL
460A F244      34150          DEFW    DELREM
460C 2C46      34200          DEFW    LISTER
460E 1E46      34250          DEFW    FINDER
4610 2746      34300          DEFW    LLISTR
               34350
4612 219F46    34400 LBLERR   LD      HL,LBERMS       ;LABEL ERROR MESSAGE
4615 COA720    34450          CALL    DISSTR
4618 AF        34500          XOR     A
4619 32E140    34550          LD      (40E1H),A       ;TURN OFF AUTO
461C C3B646    34600          JP      READY
               34650
461F 45        34700 LBERMS   DEFM    'ERROR, LABEL ASSIGNED'
4634 00        34750          DEFB    0DH
4635 00        34800          DEFB    00
               34850
               34900 ; THE FOLLOWING ROUTINES ARE INCLUDED TO MAINTAIN
               34950 ; CONSISTANCY WITH THE PROGRAM TO BE OFFERED BY INSTANT
               35000 ; SOFTWARE. THEY ALLOW THE USER TO SET (INTFLG) TO RUN
               35050 ; LABEL WITH INTERUPTS ON OR OFF.
               35100 ;THE ROUTINES ARE CALLED FROM 'CMDLBL' AND 'NOMENU'.
               35150 ; WHILE THE BASIC PROGRAM IS BEING MODIFIED, THE
               35200 ; INTERUPTS SHOULD BE OFF, HOWEVER, PEOPLE CALL
               35250 ; INSTANT SOFTWARE AND COMPLAIN THAT THEIR CLOCK
               35300 ; LOSES TIME !!
               35350
46B6 CDBC46    35400 READY    CALL    INTON           ;EI ?
46B9 C3191A    35450          JP      1A19H           ;JP TO READY
               35500
46BC 08        35550 INTON    EX      AF,AF'
46BD 3AD046    35600          LD      A,(INTFLG)
46C0 B7        35650          OR      A
46C1 2808      35700          JR      Z,INTEX
46C3 FB        35750          EI
46C4 1808      35800          JR      INTEX
46C6 08        35850 INTOFF   EX      AF,AF'
46C7 3AD046    35900          LD      A,(INTFLG)
46CA B7        35950          OR      A
46CB 2801      36000          JR      Z,INTEX
46CD F3        36050          DI
46CE 08        36100 INTEX    EX      AF,AF'
46CF C9        36150          RET
               36200
               36250
46D0 00        36300 INTFLG   DEFB    00
               36350
               36400
               36450 ; END OF INTERUPT CHECK AND SET ROUTINES
               36500
               36550
               36600 ;INITIALIZE: NOT SAVED AFTER INITIALIZATION
               36650 ; NOTE !!!  THIS INITIALIZATION ROUTINE IS DESIGNED TO
               36700 ; PROTECT LABEL IN   * LOW *  MEMORY. CHANGES MUST BE
               36750 ; MADE TO USE HIGH MEMORY.
               36800
46D1 21C441    36850 INIT     LD      HL,BLKRAM       ;POINT TO RAM
46D4 11F042    36900          LD      DE,EXBLN        ;POINT TO BYTES IN PROG.
46D7 0603      36950          LD      B,3             ;NO.BYTES TO EXCHANGE
46D9 CDF746    37000          CALL    EXCHNG          ;EXCHANGE
46DC 21B241    37050          LD      HL,FSTRAM
46DF 11B244    37100          LD      DE,EXCMDL
46E2 0603      37150          LD      B,03
46E4 CDF746    37200          CALL    EXCHNG
46E7 21D146    37250          LD      HL,INIT         ;POINT TO INIT.ROUT.
46EA 3600      37300          LD      (HL),00         ;WILL BE START
46EC 23        37350          INC     HL              ; OF BASIC PROGRAM
46ED 22A440    37400          LD      (BPRPTR),HL     ;NEW BASIC PROG.POINTER
46F0 01191A    37450          LD      BC,BREADY       ;RETURN TO READY
46F3 C5        37500          PUSH    BC
46F4 C34D1B    37550          JP      NEW             ;NEW ROUT.IN BASIC ROM
               37600
46F7 7E        37650 EXCHNG   LD      A,(BL)
46F8 08        37700          EX      AF,AF'
46F9 1A        37750          LD      A,(DE)
46FA 77        37800          LD      (HL),A
46FB 08        37850          EX      AF,AF'
46FC 12        37900          LD      (DE),A
46FD 23        37950          INC     HL
46FE 13        38000          INC     DE
46FF 10F6      38050          DJNZ    EXCHNG
4701 C9        38100 LAST     RET
               38150
               38200
46D1           38250          END     IHIT
00000 TOTAL ERRORS
```

The MICROCONDUCTOR™

the ultimate
database manager
for your TRS-80®

Compose Any Software Program
By simply answering the questions,
YOU Describe the file layouts
YOU Specify the print formats
YOU Design the update functions.
The CONDUCTOR DOES THE WORK!!!



The MICROCONDUCTOR™ directs your computer to compose, organize and summarize all information you need to solve your software and business problems.

With The MICROCONDUCTOR™, your computer will be able to compose any record-keeping software you need. In the office, The MICROCONDUCTOR™ can help with anything from accounts receivable to property management. You'll find that The MICROCONDUCTOR™ is ideal for the shop too. Let it take care of your inventory records, sales analysis, price lists, and more.

The MICROCONDUCTOR™ is not just a file manager but a true Data Base Management System suitable for both the novice and professional users.

Some of the modules of this masterpiece are:

DATA FILE—One step file creation. Just set it, and forget it.
MAINT.— Manipulate your data files with ease: add, delete, modify, scan, relocate, and more
SORT— Sort any number of fields, in any sequence, ascending or descending order.
UPDATE— Single or dual file report and update utility.

Introductory Prices

TRS-80® Model I **$249**
TRS-80® Model II **$399**

The MICROCONDUCTOR™ is power at your fingertips! Power to set up, maintain, sort, report, and update data files at whim. Just imagine: with the MICROCONDUCTOR™, you can establish a custom mailing list system in 30 minutes, accounts receivable in 2 hours, a complete business system in only a few working days. Never before has your computer been given such power!

DATA FILES—No limit on the number of records a file can have.
FIELDS—Any type (string, interger, single, double). Eight entry modes (including defaults, counting, and suppress).
REPORTS—Four ways to generate reports. Total numeric column(s). Print on any paper in any format (statements, labels, etc.).
SORT—Any field(s) in any combination (i.e. multiple-key sort). Any size file, numeric or ASCII. Ascending or descending.
MAINT.—Command anticipation. Record duplication. Direct access and sequential search.
UPDATING/MERGING—Add, subtract, multiply, divide fields. Combine results from previous calculations. Test for any condition and take action.

*Registered trade mark of Radio Shack.

MICROCOMPUTER TECHNOLOGY INCORPORATED
TELEX 676-401 TAB IRIN
Order Desk Only 800-854-7222

3304 West MacArthur
Santa Ana, CA 92704
(714) 979-9923
ALL PRICES CASH DISCOUNTED
ASK FOR OUR FREE CATALOGUE
FREIGHT FOB FACTORY

To know the joys of being on the stick, you'd better read this now.

Joystick City

Larry Suter
1643 Wersew Ave.
Livermore, CA 94550

For a few bucks and a couple hours of time you can put together a joystick, or paddle control, that plugs into your TRS-80 cassette port.

The joystick is a tool that converts variable resistances to digital signals, giving you new freedom in controlling family games. Beyond games, you can use this same, simple hardware/software combination to make your TRS-80 into a two-channel recording thermometer, a proximity sensor, a solar flux/light level meter or two ohmmeters. All this is done with a 35 cent chip, a few resistors and some capacitors.

Joystick Fundamentals

The joystick is an analog sensor that produces varying resistances.

I'm going to be using the word joystick throughout this article to describe anything that produces measurable resistences.

The basic point about joysticks is that they have two variable resistors or "pots" (potentiometers). Following normal electronic's usage, these pots are called R1 and R2. Both pots are hooked up to the control stick by a small mechanism. Because of this mechanism, when you swing the joystick left or right, only one of the pots increases or decreases its resistance.

When you move the stick forward or backward, the other pot's resistance is changed. When you move it diagonally, both pots change. To interface a joystick to your TRS-80 your computer must be able to distinguish R1 and R2. We'll use a 555 oscillator and an algorithm.

The Hardware

To make the interface there's only one fact you need to know about the 555 oscillator: It puts out a square wave as shown in Fig. 1.

The duration of high and low parts of the square wave are controlled by two resistors, R1 and R2, also shown in Fig. 1. The oscillator will be low for a length of time which is proportional to R1. Then it will be high for a time proportional to R1 + R2.

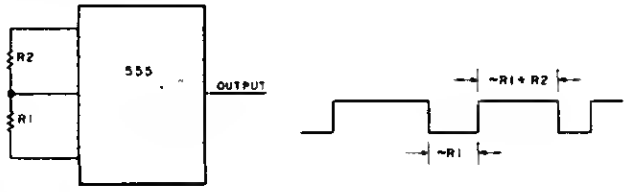


Fig. 1

```
5 CLS
7 REM CALL JOYSTICK INPUT SUBROUTINE
10 GOSUB 900
20 PRINT@0,DE,DP
25 REM SET THE JOYSTICK'S  POINT ON THE SCREEN
30 X=DE:Y=DP
40 RESET(X0,Y0):SET(X,Y):X0=X:Y0=Y
50 GOTO  10
875 REM ***********************************
885 REM JOYSTICK INPUT SUBROUTINE
895 REM WAIT FOR A TRANSITION
900 GOSUB 1000
905 REM MEASURE TIME TILL NEXT TRANSITION AND STORE IN
DE
910 GOSUB 1000
915 REM MOVE DE TO DP AND MEASURE TIME TILL NEXT TRANSI
TION
920 GOSUB 1000
925 REM MAKE SURE DE HAS THE LARGER OF DE AND DP
926 REM SWAP 'EM IF NECESSARY
930 IF DE<DP THEN TM=DE: DE=DP: DP=TM
940 DE=DE-DP
945 REM VOILA! DE IS PROPORTIONAL TO R2, DP TO  R1
950 RETURN
985 REM
986 REM ***********************************
995 REM MOVE DE TO DP
1000 DP=DE
1005 REM ZERO THE COUNTER
1010 DE=0
1015 REM RESET CASSETTE INPUT FLIP-FLOP
1020 OUT 255,0
1025 REM INCREMENT COUNTER
1030 DE=DE+1
1035 REM  READ IN PORT 255
1040 A=INP(255)
1045 REM MASK OFF BIT 7
1050 A=A AND 128
1055 REM IF BIT 7=0 THEN LOOP BACK. OTHERWISE RETURN
1060 IF A=0 GOTO 1030
1070 RETURN
```

Program Listing 1. The BASIC Algorithm

Now NRI takes you inside the world's most popular microcomputer to train you at home as the new breed of computer specialist!

NRI teams up with Radio Shack to teach you how to use, program and service microcomputers...make you the complete technician.

It's no longer enough to be just a programmer or a technician. With microcomputers moving into the fabric of our lives (over 200,000 of the TRS-80™ alone have been sold), interdisciplinary skills are demanded. And NRI can prepare you with the first course of its kind, covering the complete world of the microcomputer.

Learn At Home in Your Spare Time

With NRI training, the programmer gains practical knowledge of hardware, enabling him to design simpler, more effective programs. And, with advanced programming skills, the technician can test and debug systems quickly and easily.

Only NRI gives you both kinds of training with the convenience of home study. No classroom pressures, no night school, no gasoline wasted. You learn at your convenience, at your own pace. Yet you're always backed by the NRI staff and your instructor, answering questions, giving you guidance, and helping you over the tough spots.

Training includes TRS-80 computer, transistorized volt-ohm meter, digital frequency counter, and the NRI Discovery Lab with hundreds of tests and experiments.

(TRS-80 is a trademark of the Radio Shack division of Tandy Corp.)

Explore the TRS-80 Inside and Out

NRI training is hands-on training, with practical experiments and demonstrations as the very foundation of your knowledge. You don't just program your computer, you introduce and correct faults ...watch how circuits interact...interface with other systems...gain a real insight into its nature.

You also build test instruments and the NRI Discovery Lab, performing over 60 separate experiments in the process. You learn how your trouble-shooting tools work, and gain greater understanding of the information they give you. Both microcomputer and equipment come as part of your training for you to use and keep.

Send for Free Catalog... No Salesman Will Call

Get all the details on this exciting course in NRI's free, 100-page catalog. It shows all equipment, lesson outlines, and facts on other electronics courses such as Complete Communications with CB, TV and Audio, Digital Electronics, and more. Send today, no salesman will ever bother you. Keep up with the latest technology as you learn on the world's most popular computer. If coupon has been used, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council 179-120

```
5 REM LOAD MACHINE LANGUAGE ALGORITHM
10 AD=28223:FOR I=1 TO 51:READ A:POKE AD+I,A:NEXT
15 REM SET UP TRS-80 POINTER TO CALL MACHINE LANGUAGE A
LGORITHM
20 POKE 16526,4:POKE 16527,79
305 REM...
390 REM...
395 REM ROUTINE TO CONVERT JOYSTICK TO DIGITAL THEN PRI
NT
396 REM AND DISPLAY THE POINT
400 CLS:AD=28226:O=256:YM=.280:XM=.58
405 REM A WILL BE PROPORTIONAL TO R1, B TO R2.
410 A=USR(0):B=PEEK(AD)+O*PEEK(AD+I)
420 PRINT@0,A,B
424 REM SCALE X AND Y VALUES BEFORE PLOTTING
425 X=XM*A:Y=YM*B
427 REM MAKE SURE X AND Y AREN'T TOO LARGE FOR SET(X,Y)

428 IF Y>47THEN Y=47
429 IF X>127THEN X=127
430 RESET(X0,Y0):SET(X,Y):X0=X:Y0=Y:GOTO 410
985 REM...
990 REM...
995 REM THE MACHINE LANGUAGE ROUTINE IS CONTAINED IN TH
E
996 REM DATA STATEMENT
1000 DATA 0,0,0,0,205,36,79,205,36,79,205,36,79,213,217
,225,237,82,56,1,217,213,217,225,237,82,34,2,79,235,34,
0,79,195,154,10,217,175,211,255,17,0,0,19,219,255,230,1
20,40,249,201
```

Program Listing 2. To Load the Machine Language Algorithm

```
L2      00 00       Scratchpad to R1 and R2
L3      00 00
L1      CALL LOOP
        CALL LOOP   Collect data from cassette port.
        CALL LOOP   Store in DE and DE'.
        PUSH DE
        EXX         Find which of DE, DE' is larger
        POP HL
        SBC DE
        JR C, LA
        EXX         Put the larger in HL, smaller in DE.
LA      PUSH DE     Subtract DE from HL. Then DE has R1,
        EXX         HL has R2.
        POP HL
        SBC DE
        LDS (L3), HL
        EX          Store R1 and R2 in scratchpad.
        LD (L2), HL
        JP 0A9AH
LOOP    EXX
        XOR A       Initialize timing loop—Swap DE, DE';
        OUT FF      Reset flip-flop; zero counter
        LD DE, 00 00
LE      INC DE
        IN FF       Timing loop
        AND 80H
        JR Z, LE
        RET
```

Program Listing 3. Machine Language Algorithm with Functions Explained

The high and low durations can be input to the TRS-80 by using the cassette input plug. An alternative procedure is to use the expension connector in the rear of the TRS-80, but this approach would turn our one-night one-chip project into a week long multi-chip wire-wrap mess.

Using the cassette port is safe because the input is coupled by capacitors to the computer. This means you can mess up and dump a ±9 volt DC signal into the input and not worry about roasting the components.

Fig. 2 is a circuit diagram illustrating the interfece of the 555 oscilletor to the cassette input port. The left side of the figure is the oscillator you'll build. The right side of the diagram rapresents the parts of the TRS-80 which are important to the interface. *A* more complete (completely confusing?) diegram is shown in the back of the *TRS-80 Microcomputer Technical Reference Manual.*

Cassette Port Explained

Consider the diagram of the TRS-80 in Fig. 2. Low level voltage pulses entering your computer through the cassette ear jack go into the signal conditioner.

This conditioner converts these pulses (about 0.2 volts) or either polerity into pulses which can SET the flip-flop. For example, if the output is 0, then the pulse makes it a 1. If the output is 1, it stays 1.

Fig. 2 also shows some bipoler pulses which will SET the flip-flop.

The output of this flip-flop can be read as the eighth bit of a data word. We will read this whenever an INP(255) BASIC function, or its machine language equivalent, is executed. The path of data into your machine begins at the cassette jack, moves to the signal conditioner, to the flip-flop end to port 255 where it's read.

It's important to know that once that flip-flop is SET, it stays SET—until it is RESET.

If you want the computer to be aware of more than one pulse, you must RESET the flip-flop each time it is SET. To RESET it, send an OUT 255,0 BASIC command or its machine language equivalent.

Remember that the output of the 555 oscillator is controlled by the variable resistors R1 and R2, but this square wave output is not directly into the TRS-80. First it passes through a capacitor which produces bipolar pulses several microseconds long on the cassette input port so that every time the 555 oscillator makes a transition, up or down, the flip-flop will be set.

By measuring the time between successive settings and using an algorithm, we can find R1 and R2 or, rather, generate numbers which are proportioned to R1 and R2.

The Algorithm

We'll measure time between

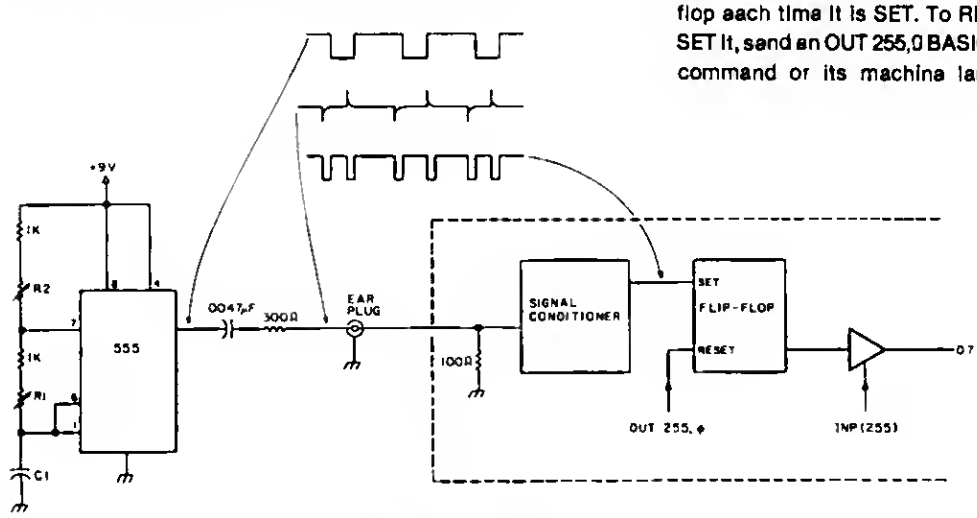


Fig. 2

1 Joystick w/100 K pots
1 555 chips
2 1K resistors
1 0.1 microfarad capacitor
1 1.0 microfared capecitor (optional)
1 9-volt battery
1 battery chip
1 miniature phone jack
1 300 Ω resistor
1 4.7 nF capacitor

Parts List

Combine accurate flight characteristics with the best in animation graphics and you'll have SubLOGIC's

T80-FS1 Flight Simulator

for the TRS-80

SubLOGIC's T80-FS1 is the smooth, realistic simulator that gives you a real-time, 3-D, out-of-the-cockpit view of flight.

Thanks to fast animation and accurate representation of flight, the non-pilot can now learn basic flight control, including take-offs and landings! And experienced pilots will recognize how thoroughly they can explore the aircraft's characteristics.

Once you've acquired flight proficiency, you can engage in the exciting British Ace 3-D Aerial Battle Game included in the package. Destroy the enemy's fuel depot while evading enemy fighters.

Computer and aviation experts call the T80-FS1 a marvel of modern technology. *You'll* simply call it *fantastic!*

Special Features:
- 3 frame-per-second flicker free animation
- Maximum transfer keyboard input
- Constant feedback cassette loader

Hardware Requirements:
- Radio Shack TRS-80, Level 1 or 2
- 16K memory
- *Nothing else!*

Only $25

See your dealer or order direct. For direct order, include $1.25 and specify UPS or first class mail. Illinois residents add 5% sales tax. Visa and Mastercard accepted.

subLOGIC
Distribution Corp.
Box V, Savoy, IL 61874
(217) 359-8482

✓150

! STOP !

OUT OF MEMORY ERRORS
with
VARKEEP
—A—
MEMORY MANAGEMENT UTILITY

- **Chain programs without losing variables**
- **Change the amount of string space without losing variables**
- **Redimension arrays**
- **Reclaim memory from variables no longer needed.**
- **Protect variables from deletion when making program changes**

ADD
FOUR POWERFUL COMMANDS
TO YOUR BASIC

LEVEL II OR DOS

$16.95 diskette . . . . . tape $14.95

● 517/485-0344 517/487-3358 ●
COD/MC/VISA
MC/VISA CARD 4% ADDITIONAL CHARGE
To order your copy, Call or Send name, address and $16.95 for disk or $14.95 for tape, plus .75¢ shipping to:

✓138

THE ALTERNATE SOURCE
1806 ADA STREET
LANSING, MI 48910

Let Your TRS-80® Teach You ASSEMBLY LANGUAGE

Tired of buying book after book on assembly language programming and still not knowing your **POP** from your **PUSH**?

REMSOFT proudly announces a more efficient way, using your own TRS-80®, to learn the fundamentals of assembly language programming --at YOUR pace and at YOUR convenience.

Our unique package, "**INTRODUCTION TO TRS-80® ASSEMBLY PROGRAMMING**", will provide you with the following:

- Ten 45-minute lessons on audio cassettes.
- A driver program to make your TRS-80® video monitor serve as a blackboard for the instructor.
- A display program for each lesson to provide illustration and reinforcement for what you are hearing.
- A textbook on TRS-80® Assembly Language Programming.
- Step-by-step dissection of complete and useful routines to test memory and to gain direct control over the keyboard, video monitor, and printer.
- How to access and use powerful routines in your Level II ROM.

This course was developed and recorded by Joseph E. Willis and is based on the successful series of courses he has taught at Meta Technologies Corporation, the Radio Shack Computer Center, and other locations in Northern Ohio. The minimum system required is a Level II, 16K RAM.

REMASSEM-1 only $69.95

REMSOFT, Inc.
571 E. 185 st.
Euclid, Ohio 44119
(216)531-1338

Include $1.50 for shipping and handling
Ohio residents add 5½% sales tax
TRS-80® is a trademark of the Tandy Corp.

✓70

---

*TRS-80 LV II, 16K

% STAT-BALL %
Baseball Simulation

% STAT-BALL % is not an arcade type game, but a comprehensive, 3 part statistical baseball simulation package that uses real life pitcher and batter statistics to simulate game play.

During the game, 12 performance categories are computed and displayed for batters, 10 for pitchers. These stats reflect the player's game performance and will follow closely a player's actual ability. Stats may be viewed on a per game basis or kept for many games.

Agonize when your best power hitter drops a routine fly, feel the thrill of victory when he hit's a deep fly over the wall. All action is described as it happens to add tension and excitement to this engrossing game.

YOU'RE THE MANAGER! Select your own lineup, make all types of substitutions, call your own strategy and live with it!! Play modern, old timer, or mixed teams.

This package consists of 3 programs:

(1) GAME
(2) RECORDS (Sorts up to 150 players into 14 categories based on relative performance
(3) WORKUP/EDIT (Establish, edit, or mix players)

Features include
- Solitaire or head to head competetion
- Batters ability ratings include bunting running, and errors
- Pitcher tiring factor
- 3 offensive and defensive strategy options
- Full 25 man rosters
- Printer option (32 chr/in or more)
- Top quality tape
- Detailed instructions
- FREE '79 Pirates and Orioles for series replay

For prompt, 1st class delivery send $25.00 check or money order to

J.L.S. SOFTWARE ✓315
P.O. Box 10385
Chicago, Illinois 60610

(We pay shipping and handling)

(*Trademark TANDY CORP.)

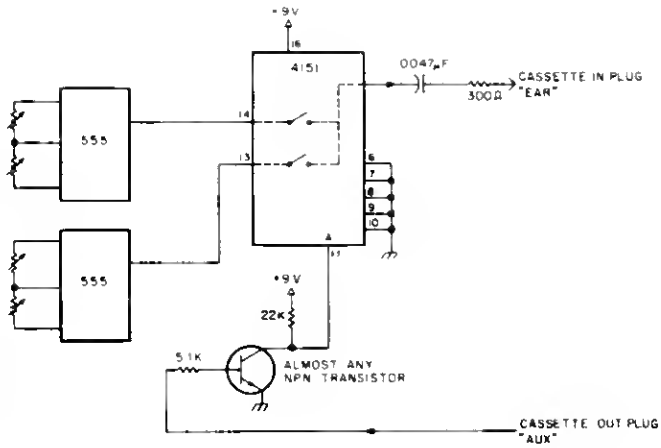
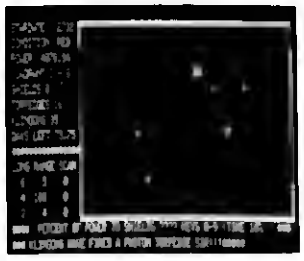


Fig. 3

```
5 REM LOAD MACHINE LANGUAGE ALGORITHM AND SET-UP USR(0)
    POINTER
10 AD=20223:TP=16526:FORI=1 TO 51:READ A:POKE AD+I,A:NE
    XT
15 POKE TP,4:POKE TP+1,79
18 REM MACHINE LANGUAGE ALGORITHM IS IN DATA STATEMENT
20 DATA 0,0,0,0,205,36,79,205,36,79,205,36,79,213,217,2
    25,237,82,56,1,217,213,217,225,237,82,34,2,79,235,
    34,0,79,195,154,10,217,175,211,255,17,0,0,19,219,2
    55,230,120,40,249,201
85 REM...
90 REM...
95 REM...SLALOM GAME FOLLOWS
98 REM Y(I) IS THE VERTICAL LOCATION OF GATE # I
99 REM WE START OFF WITH NO GATES DISPLAYED
100 DIM Y(3),X(3):CLS:N=34:FORI=1TO3:Y(I)=45:NEXT
115 REM BEGIN MAIN LOOP
117 REM     DY IS THE VERTICAL GATE MOVEMENT PER LOOP.
118 REM     THIS IS CONTROLLED BY YOUR JOYSTICK.
120 DY=PEEK(AD+3)/48
122 GOSUB 500
130 REM BEGIN GATE MOVEMENT LOOP
139 REM I IS THE GATE INDEX, WE COUNT DOWN IN THIS LOOP
    .
140 I=2
148 REM THE POINTS (J,K) AND (M,K) FORM THE GATE. IF IT
    S
149 REM OFF THE SCREEN THEN WE GOTO 300 TO GET A NEW GA
    TE
150 K=Y(I):J=K(I):M=J+10:IF K>43 THEN 300
154 REM GET SKIIER'S POSITION. SET HIS LOCATION.
155 GOSUB 500
159 REM MOVE GATE # I
160 L=K+DY:Y(I)=L
170 RESET(J,K):RESET(M,K):SET(J,L):SET(M,L)
177 REM SEE IF SKIIER CROSSED THE GATE'S Y-POSITION.
178 REM IF SO THEN GOSUB TO SEE IF HE PASSED THRU THE G
    ATE
180 IF K<N AND L>N GOSUB400
195 REM NEXT GATE'S INDEX, OR LOOP TO 120 IF WE'VE MOVE
    D ALL
196 REM THE GATES. I>1 FOR 2 GATES. I>0 FOR 3.
200  I=I-1:IFI>0 GOTO 150
210 GOTO120
290 REM 1-.90 IS THE PROBABILITY THAT A GATE WILL BE GE
    NERATED.     ADJUST THIS PARAMETER TO SUIT YOURSEL
    F.
300 RESET(J,K):RESET(M,K):IF RND(0)<.90 GOTO 200
304 REM NG COUNTS THE NUMBER OF GATES GENERATED
305 NG=NG+1
309 REM GENERATE NEW GATE'S X-POSITION.
310  Y(I)=0:K(I)=117*RND(0):GOTO 200
390 REM SEE IF SKIIER PASSED THRU THE GATE
400 IF S>J AND S<N RETURN
409 REM NM COUNTS # OF GATES MISSED. MISS 5 AND YOU'RE
    DONE.
410 NM=NM+1:IFNM<5THEN:PRINT@0,NM:RETURN
415 REM SCORING ALGORITHM
420 CLS:SC=RND(10000)
420 REM PRINT NOW SKIIER DID
430 PRINT"# OF GATES",NG
440 PRINT"#  MISSED",NM
450 FORI=1TO1000:NEXT
455 REM PRINT SCORE IN BIG LETTERS
460 CLS:OUT255,0:PRINT"S C O R E = ",SC
470 FOR I=1TO3000:NEXT:CLS
490 REM THIS SUBROUTINE READS THE JOYSTICK AND DISPLAYS
    THE SKIIER'S POSITION.
500 S=USR(0)/2-15:IFS>127THENS=127
505 IF S<0 THEN S=0
510 RESET(S0,N):SET(S,N):S0=S:RETURN
```

Program Listing 4. Slalom

SOFTWARE → TRS-80 ← SOFTWARE

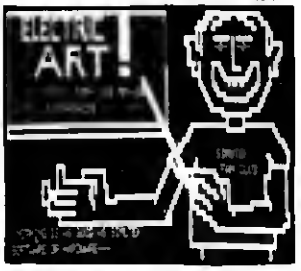


PACKAGE ONE INCLUDES: GRAPHIC-TREK "2000" — This full graphics, real time game is full of fast, exciting action! Exploding photon torpedoes and phasers fill the screen! You must actually navigate the enterprise to dock with the giant space stations as well as to avoid klingon torpedoes! Has shields, galactic memory readout, damage reports, long range sensors, etc! Has 3 levels for beginning, average, or expert players! ★ **INVASION WORG** — Time: 3099, Place: Earth's Solar System Mission: As general of Earth's forces, your job is to stop the Worg invasion and destroy their outposts on Mars, Venus, Saturn, Neptune, etc! Earth's Forces: Androids — Space Fighters — Lazer Cannon — Neutrino Blasters! Worg Forces: Robots — Saucers — Disintegrators — Proton Destroyers! Multi level game lets you advance to a more complicated game as you get better! ★ **STAR WARS** — Manuever your space fighter deep into the nucleus of the Death Star! Drop your bomb, then escape via the only exit. This graphics game is really fun! May the Force be with you! ★ **SPACE TARGET** — Shoot at enemy Ships with your missiles. If they eject in a parachute, capture them — or if you're cruel, destroy them! Full graphics, real time game! ★ **SAUCERS** — This fast action graphics game has a time limit! Can you be the commander to win the distinguished cross! Requires split second timing to win! Watch out!

ONLY 14.95

PACKAGE TWO INCLUDES: CHECKERS 2.1 — Finally! A checkers program that will challenge everyone! Expert as well as amateur! Uses 3-ply tree search to find best possible move. Picks randomly between equal moves to assure you of never having identical games. ★ **POKER FACE** — The computer uses psychology as well as logic to try and beat you at poker. Cards are displayed using TRS-80's full graphics. Computer raises, calls, and sometimes even folds! Great practice for your Saturday night poker match! (Plays 5 card draw). ★ **PSYCHIC** — Tell the computer a little about yourself and he'll predict things about you, you won't believe! A real mind bender! Great amusement for parties. ★ **TANGLE MANIA** — Try and force your opponent into an immobile position. But watch out, they're doing the same to you! This graphics game is for 2 people and has been used to end stupid arguments. (And occasionally starts them!) ★ **WORD SCRAMBLE** — This game is for two or more people. One person inputs a word to the computer while the others look away. The computer scrambles the word, then keeps track of wrong guesses.

ONLY 14.95

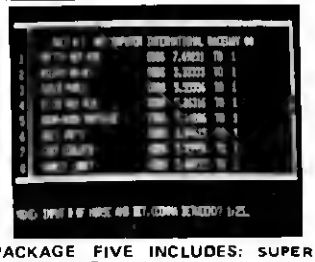


PACKAGE THREE INCLUDES: POETRY — This program lets you choose the subject as well as the mood of the poem you want. You give TRS-80 certain nouns or names, then the mood, and it does the rest! It has a 1000-word + vocabulary of nouns, verbs, adjectives and adverbs! ★ **ELECTRIC ARTIST** — Manual: draw, erase, move as well as, Auto: draw, erase and move. Uses graphics bits not bytes. Saves drawing on tape or disk! ★ **GALACTIC BATTLE** — The Swineus enemy have long range phasers but cannot travel at warp speed! You can, but only have short range phasers! Can you blitzkrieg the enemy without getting destroyed! Full graphics — real time! ★ **WORD MANIA** — Can you guess the computer's words using your human intuitive and logical abilities? You'll need to, to beat the computer! ★ **AIR COMMAND** — Battle the Kamikaze pilots. Requires split second timing. This is a FAST action arcade game.

ONLY 14.95

PACKAGE FOUR INCLUDES: LIFE This Z-80 machine language program uses full graphics! Over 100 generations per minute make it truly animated! You make your starting pattern, the computer does the rest! Program can be stopped and changes made! Watch it grow! ★ **SPACE LANDER** — This full graphics simulator lets you pick what planet, asteroid or moon you wish to land on! Has 3 skill levels that make it fun for everyone. ★ **GREED II** — Multi-level game is fun and challenging! Beat the computer at this dice game using your knowledge of odds and luck! Computer keeps track of his winnings and yours. Quick fast action. This game is not easy! ★ **THE PHARAOH** — Rule the ancient city of Alexandria! Buy or sell land. Keep your people from revolting! Stop the rampaging rats. Requires a true political personality to become good! ★ **ROBOT HUNTER** — A group of renegade robots have escaped and are spotted in an old ghost town on Mars! Your job as "Robot Hunter" is to destroy the pirate machines before they kill any more settlers! Exciting! Challenging! Full graphics!

ONLY 14.95



PACKAGE FIVE INCLUDES: SUPER HORSERACE — Make your bets just like at the real racetrack! 8 horses race in this spectacular graphic display! Up to 9 people can play! Uses real odds but has that element of chance you see in real life! Keeps track of everyone's winnings and losses. This is one of the few computer simulations that can actually get a room of people cheering! ★ **MAZE MOUSE** — The mouse with a mind! The computer generates random mazes of whatever size you specify, then searches for a way out! The second time, he'll always go fastest route! A true display of artificial intelligence! Full graphics, mazes & mouses! ★ **AMOEBA KILLER** — You command a one man submarine that has been shrunken to the size of bacteria in this exciting graphic adventure! Injected into the president's bloodstream, your mission is to destroy the deadly amoeba infection ravaging his body! ★ **LOGIC** — This popular game is based on Mastermind but utilizes tactics that make it more exciting and challenging — has 2 levels of play to make it fun for everyone. ★ **SUBMARINER** — Shoot torpedoes at the enemy ships to get points. Fast action graphics, arcade type game is exciting and fun for everybody!

ONLY 14.95

HARDWARE → TRS-80 ← HARDWARE

MICRO SPEED

Upgrade your "slow" TRS-80 to a **SUPER FAST MACHINE!!** (2.66 MHZ) over **50% FASTER!** Some of the features:

Auto turn-off during cassette or disk access. (This means NO lost programs EVER!) (Turns back on automatically too!) MANUAL control. (Unit may be turned on or off at any time. Yes even during program execution!) Keyboard indicator light "blinks" when micro-speed is on. Stops blinking when off! **Don't wait for SARGON II or any other program!!!** Comes with easy to follow instructions. (Some soldering required.) **OR** take to your local computer store or TV-Appliance Center for quick installation. (5-10 minutes!!!) Works with any model, TRS-80.

ONLY 29.95 complete

MICRO BEEP

Simple hook up: Just plug cassette remote jack into unit.

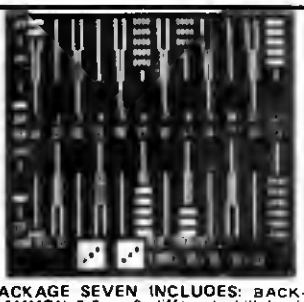
EASILY CONTRDLLED FROM BASIC:

OUT 255,4 = on
OUT 255,0 = off

MICRO-BEEP make games more fun as well as provide useful sound output for professional applications!

Works with Any Model I TRS-80

ONLY 15.95 complete



PACKAGE SEVEN INCLUDES: BACKGAMMON 5.0 — 2 different skill levels make this game a challenge to average or advanced players. FAST (15 second avg) Looks for best possible move to beat you! FANTASTIC GRAPHICS. Plays doubles and uses international rules ★ **SPEED READING** — Increases your reading speed. Also checks for comprehension of material. Great for teenagers and adults to improve reading skills. ★ **PT 109** — Drop depth charges on moving subs. Lower depths get higher points in this fast action graphics game. ★ **YAHTZEE** — Play Yahtzee with the computer. This popular game is even more fun and challenging against a TRS-80! ★ **WALL STREET** — Can you turn your $50,000 into a million dollars? That's the object of this great game. Simulates an actual stock market!

ONLY 14.95

PACKAGE SIX INCLUDES: 20 HOME FINANCIAL PROGRAMS — Figures amortization, annuities, description rates, interest tables, earned interest on savings and much, much more. These programs will get used again and again. A must for the conscientious, inflation minded person.

ONLY 14.95

SIMUTEK

✓19

Exceptional Products through Research & Imagination

Send Check, Money Order or Bank Card No. orders to:

SIMUTEK
P.O. Box 13687
Tucson, AZ 85732

FREE Postage and Handling

Master Charge

Call Toll Free
(800) 528-1149
(C.O.D. $3 extra)

Visa

Same Day Shipment on Bank Cards, Money Orders & C.O.D.

All Tape Programs Require a Minimum of 16K Level 2
Packages Available on Diskette (32K System) $4.25 Extra
3 or More Packages Get 10% Discount

Dealer Inquiries Invited

TRS-80 IS A REGISTERED TRADEMARK OF TANDY CORP.

SMALL SOFTWARE SYSTEM PRODUCTS FOR THE TRS-80 SMALL SOFTWARE SYSTEM

NEW!

WHISTLER: HOME CONTROLLER INTERFACE - $34.95. New hardware product that controls lights, appliances, computer peripherals, darkroom timers and other 115 volt devices anywhere in your house! Software controlled by cassette cable. Use with Sears or BSR Home Control System with ultrasonic option. Assembled, tested, self-contained, and includes Basic software.

TRS-80 DISK & OTHER MYSTERIES - $22.95, H.C. Pennington. Best disk book we've seen! Directory secrets, file formats, damaged disk recovery, etc.

LEARNING LEVEL II - $15.95, D.A. Lien. Learn Level-2 like you did Level-1, step by step. Same author and style as Level-1 manual. Super new book!

UTILITIES

RSM-2: MACHINE LANGUAGE MONITOR FOR 16K TRS-80'S - $26.95
RSM-2D: THREE VERSIONS OF RSM-2 FOR DISK SYSTEMS - 29.95
RSM-2 RELOCATOR: PUT RSM-2/2D ANYWHERE IN MEMORY - 9.95

Machine Language monitors with Z-80 disassembler! HEX and ASCII memory dumps; EDIT, MOVE, EXCHANGE, VERIFY, FILL, ZERO, TEST, or SEARCH memory, read/write SYSTEM tapes, enter BREAKPOINTS, PRINT with TRS232 or Centronics, read/write disk sectors directly! RSM-2 tape loads at top of 16K LEVEL I or II; RSM-2D disk includes 3 versions for 16K, 32K and 48K.

DCV-1: CONVERT SYSTEM PROGRAMS TO DISK FILES -$9.95. Execute Adventure, Air Raid, RSL-1, ESP-1, T-BUG, etc. from disk, even if they interfere with TRSDOS! New version works with TRSDOS 2.3.

BASIC-1P: LEVEL-1 BASIC WITH PRINTING! - $19.95. Run any LEVEL-I BASIC program on your 16K Level-2. PLUS LPRINT and LLIST with our TRS232 or Centronics. Furnished on tape; can be used from disk.

MACHINE LANGUAGE GAMES

AIR RAID, BARRICADE or RNL-1: - $10.00 each, all 3 for $25.00

AIR RAID: A super shooting gallery; our most popular game. Ground based missile launcher shoots high speed aircraft! Hours of fun!

BARRICADE: "BREAKOUT" for the TRS-80! Break through 5 walls with high-speed ball and keyboard controlled paddle! 96 different options!

RSL-1: Enter patterns with repeating keyboard! Save patterns on tape (4 furnished). Play John Conway's LIFE. FAST - about 1 second per generation!

MODEL-II TRS-80'

CP/M™ VERSION 2.0 FOR THE MODEL-II - $170.00. Latest version from Digital Research. Runs both single and double density disks! "Standard" version runs nearly any CP/M software, including Cobol, Fortran, C-Basic, M-Basic, business and accounting packages, etc. Hundreds of programs available!

RSM II: ENHANCED RSM MONITOR FOR THE MODEL-II - $39.95. Relocatable version of RSM-2D plus screen editor for modifying either memory or disk sectors in both Hex and ASCII, split screen scrolling, and formatted serial or parallel printing. Sold on self-booting disk; directions to save as TRSDOS file.

PROFESSIONAL SOFTWARE

THE ELECTRIC PENCIL FOR THE TRS-80: TAPE-$99.95, DISK-$150.00. Popular video editor for creating and saving text files. Prints formatted copy with right justification, page titling & numbering, etc. Upper case only, or lower case with modification. 16K Level-1 or 2 (tape).

CP/M™ OPERATING SYSTEM FOR THE MODEL-I - $145.00. The 8080/Z80 "Software Bus for the Model-1 TRS-80. Includes TRS232 and RS-232-C software, lower-case support, debounce, DCV-2 and other unique utilities. Allows use of many available programs written for CP/M.

PRINTER SUPPORT

TRS232 PRINTER INTERFACE - $49.95 ($59.95 after June 30). Assembled & tested printer interface for RS232 or 20-mil current loop printers. Expansion interface not required. Print from Level-II BASIC, CP/M, BASIC-1P, ELECTRIC PENCIL, etc. Standard cassette software included. Add $2.00 for shipping.

TRS232 "FORMATTER" SOFTWARE PACKAGE - $14.95. Adds page and line length control, printer pause, "smart" line termination, etc. to TRS232.

RSM232: Adds RS-232-C capability to RSM-2/2D monitors - **$14.95**
PEN232: RS-232-C for cassette version Electric Pencil - **$14.95**
EBT232: TRS232 and RS-232-C for tape version of EDTASM - **$14.95**

OTHER PRODUCTS FOR THE TRS-80

ESP-1: $29.95. Assembler, Editor, Monitor (8080 mnemonics)
LST-1: 8.00. Listing of Level-1 BASIC with some comments

**CP/M tm Digital Research, Inc. *TRS-80 tm Tandy Corp.
See your dealer or order direct. Calif. Residents add 6% tax

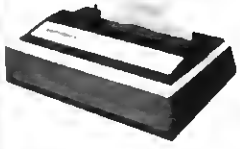
SMALL SYSTEM SOFTWARE ■ P.O. BOX 366 ■ NEWBURY PARK, CA 91320

SMALL SYSTEM SOFTWARE ■ P.O. BOX 366 ■ NEWBURY PARK, CA 91320

PRINTERS & CRT'S From Orange Micro

CENTRONICS 737 (RADIO SHACK LINE PRINTER IV)

Word Processing Print Quality

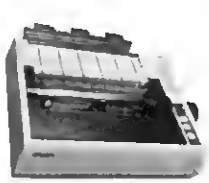


• 18 x 9 dot matrix; suitable for word processing • Underlining • proportional spacing • right margin justification • serif typeface • 50/80 CPS • 9½" Pin Feed/Friction feed • Reverse Platen • 60/132 columns

CENTRONIC 737-1 ..... (List $995) **$Call**

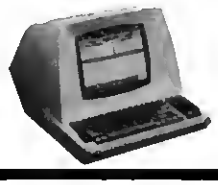
EPSON MX80

Low-Priced
Professional Print Quality



• 9 x 9 dot matrix • Lower case descenders • 60 CPS • Bidirectional, Logic seeking • 40, 66, 80, 132 columns per line • 64 special graphic characters: TRS-80 Compatible • Forms handling • Multi-pass printing • Adjustable tractors

EPSON MX80 ..... (List $645) **$Call**



TELEVIDEO CRT'S
PRICES SLASHED!

TVI 912C
TVI 920C
} Please Call Toll Free Prices are too low to advertise

PRINTERS

ANACOM 150 150 CPS, wide carriage, 9 x 9 dot ..... (List $1350) **$ Call**
CENTRONICS 737 Text processing dot matrix (Radio Shack LP IV) ..... **$ Call**
CENTRONICS 730 (Radio Shack Line Printer II) **639**
BASE 2 800B graphics printer ..... (List $699) **$649**
OKIDATA MICROLINE 60 ..... (List) $800) **599**
NEL 5530-5 letter quality, RO, parallel, tractors ..... (List $2970) **$2549**
MALIBU Dot graphics, 132 Col, Letter quality ..... **$ Call**
PAPER TIGER IDS 440 w/graphics & 2K buffer ..... (List $1094) **939**
QUME 5/45 Typewriter quality ..... (List $2905) **2499**

INTERFACE EQUIPMENT

APPLE II — BASE 2 parallel graphics interface board **160**
SSM AIO BOARD Serial/Parallel interface board ..... (List $225) **199**
TRS-80 CABLES expansion interface or direct ..... **$ Call**

TOLL FREE (800) 854-8275

CA, AL, HI (714) 630-3322

Call for FREE CATALOG

Phone orders WELCOME. Same day shipment for VISA, MASTER CHARGE, and AMERICAN EXPRESS. Personal checks require 2 weeks to clear. Add 3% for shipping and handling. California residents add 6%. Manufacturer's warranty included. Prices subject to revision.

Orange Micro, Inc.
3148 E. La Palma, Suite E
Anaheim, CA 92806

```
L2      00 00
L3      00 00
L1      CALL LOOP
        CALL LOOP
        CALL LOOP
        PUSH DE          See Program Listing 3 for description
        EXX
        POP HL
        SBC DE
        JR C, LA
        EXX
LA      PUSH DE
        EXX
        POP HL
        SBC DE
        PUSH HL
        LD IX,(L3)       Stashes R2 on stack. Averages and stores R1.
        CALL AVRG
        LD(L3),HL
        POP DE
        LD IX,(L2)       Retrieves R2. Averages and stores R2.
        CALL AVRG
        LD (L2),HL
        LD B,04H
        LD A,7FH
L8      SRL H
        RR L
        DJNZ L8          Puts the smaller of R2/18 or
        CPL              127 into HL. Returns HL as the
        JR NC, LC        value of the USR(0) function.
        LD L,A
LC      JP 0A9AH
AVRG    PUSH IX
        POP HL           Averaging subroutine.
        LD B,03H         Divides N by 18.
L6      SRL H
        RR L
        DJNZ L6
        SBC DE           N/8 - R into HL
        LD A,L
        JR C,L7          IF ABS(N/8 - R)>4
        NEG
L7      CP 04
        JR C, L8         THEN GOTO L8
        EX
        PUSH IX
        POP HL           ELSE HL = 7/8*N + R
        SBC DE
        RET
L8      EX
        LD B,03H
L9      SLA,L            HL = R*8
        RR H
        DJNZ L9
        RET
LOOP    EXX
        XOR A
        OUT FF           Same timing loop as Program Listing 3.
        LD DE 00 00
LD      INC DE
        IN FFH
        AND 80H
        JR Z, LD
        RET
```

Program Listing 5

```
10 AD=28227:TP=16526:FORI=1TO115:READA:POKEAD+I,A:NEXT:
   POKETP,4:POKETP+1,79
15 DATA 205,184,79,205,184,79,205,184,79,213,217,225,23
   7,82,56,1,217,213,217,225,237,82,229,221,42,2,79,2
   05,65,79,34,2,79,209,221,42,0,79,205,65,79,34,0,79
   ,6,4,62,127,203,60,203,29,16,250,189,48,1,111,195,
   154,18,221,229,225,6,3,203,60,203,
20 DATA 16,250,237,82,125,48,2,237,68,254,4,48,7,235,22
   1,229,225,237,82,201,235,6,3,203,37,203,20,16,250,
   201,217,175,211,255,17,0,0,19,219,255,230,128,40,2
   49,201
100 CLS:Q=256: AD=28224
105 XM=.85:YM=.82
110 S=USR(0):X=PEEK(AD-MQ*PEEK(AD+1):Y=PEEK(AD+2)+Q*PEE
    K(AD+3)
120 PRINT@0,S,X,Y:X=XM*X:Y=YM*Y
125 IF X>127 THEN X=127
126 IF Y>47 THEN Y=47
130 RESET(X0,Y0):SET(X,Y):X0=X:Y0=Y
140 GOTO 110
```

Program Listing 6. Data statement

transitions with a timing loop. After locating a transition, zero a counter; RESET the input flip-flop with an OUT 255,0 instruction; and ENTER the loop. The loop is diagrammed in the Flow Chart.

When a transition has occurred the counter has a number proportional to either R1 or R1 + R2.

(You're half way. You'll be finished when you store the counter's value; re-zero the counter; RESET the flip-flop; and repeat the loop.)

If you had R1 the first time, this time you'll have R1 + R2 (and vice versa). At this point you have two numbers, and you don't even know which is which. The algorithm takes advantage of the fact that R1 + R2 is obviously larger than R1.

Program Listing 1 is a BASIC form of the algorithm which establishes the joystick's settings. It also PRINTs the numbers proportional to R1 and R2 and displays them as points on the screen.

This BASIC program has one drawback: It's dreadfully slow. To use it, C1 of Fig. 2 has to be large. Ten microfarads is barely enough with 100K pots on the joystick. Even though it's slow, try it before moving on to the machine language version.

Machine Language Algorithm

Program Listing 2 automatically loads a machine language version of the joystick algorithm into memory. The program sets up the appropriate TRS-80 pointer so that when you execute the A=USR(0) function, you will convert the joystick settings.

Type the listing and double check the DATA statement. Since the program's more than 1000 times faster than the BASIC algorithm of Program Listing 1, you'll need to make C1 about 0.1 microfarads.

A, the value returned by A=USR(0), will be proportional to R1. Since USR(0) returns only one value, we've chosen to store the number proportional to R2 as a two-byte integer in locations 20224 and 20225.

```
B = PEEK(20224) + 256*PEEK(20225)
```

Line 10 of Program Listing 2 recovers R2. If you make C1 equal to 0.1 microfarad and put a large resistor (200K to 500K) in parallel with R2, R2 will be less than 255. This means that the number proportional to R2 is stored entirely in 20224 and the much quicker B = PEEK(20224) will recover it for you.

The rest of Program Listing 2 is a demo which shows you the type of coding you can use to let the joystick move a dot all over your monitor. Note that you'll have to multiply A and B by scale factors to make the full travel of the joystick's dot comparable to the screen size. You may have to juggle these parameters a bit.

Program Listing 3 is an assembly language listing of the algorithm in Z-80 mnemonics, accompanied by an explanation of each function.

Slalom

Program Listing 4 lets you play an arcade-type joystick game on your TRS-80. The object of Slalom is to ski a point through a series of gates which move down from the top of the screen. You move the skier around the screen with the joystick. Moving the joysticks forward or backward controls the speed of the gates.

Since I'm more interested in building and interfacing peripherals than in BASIC programming, I'm sure there's room for you to improve Slalom.

I kind of like Slalom—not only because it demonstrates how to utilize your joystick, but also because it demonstrates a useful way of interfacing a machine language routine with a BASIC program. When you run the game, the first thing the program does is to load the algorithm into memory and set up the USR(0) pointer.

The DATA statement of this figure contains exactly the same algorithm as Listing 2 and 3. When you write your own joystick games you might want to use lines 1 to 20 to start your routine.

The Loophole Algorithm

One problem with increasing the joystick's speed is that we end up with about one percent jitter in its settings. When running Program Listing 2, this jitter causes the point to occasionally hop back and forth on the screen. This causes no trouble for a simple game such as Slalom. But it would cause problems if you were using the joystick to do something more precise, such as manipulate a cursor.

You can eliminate the jitter with an auxiliary algorithm. When you measure A=USR(0), where A is proportional to R1, set:

```
N = 7/8*N + A
```

This will make N the running average of the last eight values we measured. N will have little jitter.

The drawback of this averaging technique is that you have to wait while N establishes its new value when you move the joystick and suddenly change A.

To get around this, include a loophole in the software so that when you move the joystick, N will first change rapidly and then start averaging to produce low jitter values. The loophole algorithm is:

```
IF ABS(N/8 - A)<3 THEN N = .875*N + A
ELSE N = 8*A
```

Try this in the routine in Listing 2. You'll have to change line 425 to read X = XM*N before doing the SET(X,Y).

The same technique can be used to eliminate any up and down jitter, too.

If you need to eliminate the jitter and want to do it quickly, use the machine language algorithm of Program Listing 5. This algorithm reads the joystick, finds numbers proportional to R1 and R2 and averages them over the last eight reads. It's just like the BASIC coding, except it's much faster.

Program Listing 6 shows a DATA statement which you can use to automatically load the algorithm into memory. This algorithm will store the value of N, proportional to R1, as a two-byte integer in locations 20224 and 20225. Following the usual convention, 20224 has the least significant byte and 20225 has the most significant.

The averaged value proportional to R2 is stored in 20226 and 20227. In addition to storing the numbers, when you execute this algorithm with S = USR(0), S will be proportional to R2.

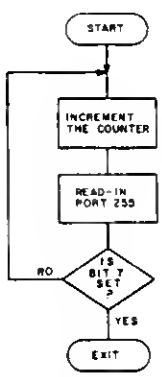
Beyond a Single Joystick

Fig. 3 is a circuit which lets you use two joysticks. The key trick here is to use the cassette-out plug to select the joystick. We "amplify" and invert the cassette-out signal (0V to 0.9V) with a transistor.

The signal goes into select pin A of a 4051 analog switch. A zero at input A of the 4051 hooks up one of the 555s to the capacitor. A one on A hooks up the other.

To select one joystick, you must issue an OUT 255,2 instruction which puts zero volts on the cassette-out plug. To select the other joystick, you issue an OUT 255,1 instruction. This puts 0.9V on the cassette-out plug. You must do this before calling the algorithm. You must also do this, consistently, within the algorithm.

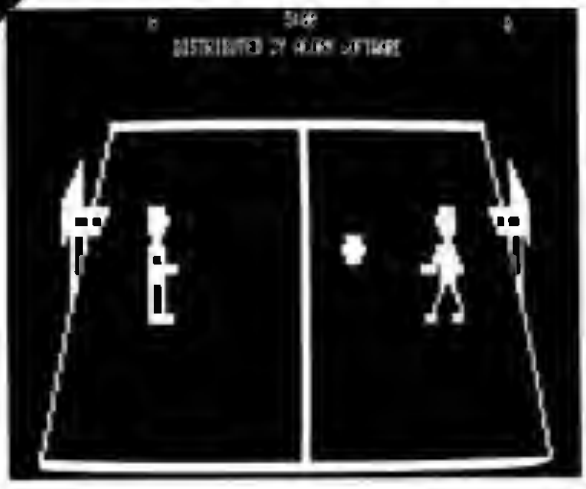
Games are fun, but this system and your imagination can make much more than games. With minor modifications you can also use this hardware/software system to make relatively sophisticated instruments and controls. ■



Flowchart

FOR TRS-80*

NEW! BASKETBALL



by John Allen

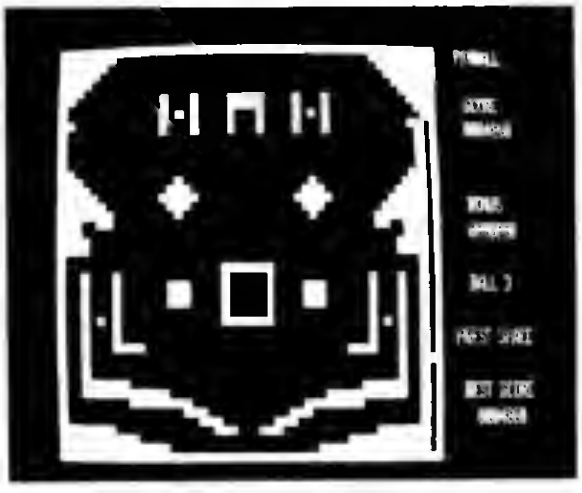
New machine language action game, with sound, from the author of the acclaimed "PINBALL!"

You have to be fast to keep up with the action as you try to outscore your opponent in five minutes of one-on-one basketball. Compete against a friend or your computer.

Steal the ball, duck around your opponent and slant toward the basket for a lay up! The graphics are based on a 3-dimensional depiction of a basketball court, and ball dribbling sounds add to the realism. It's all there but the cheers—so real you'll wonder how the ball keeps from coming through the screen of your TRS-80! Dribble, Dribble!

Available for Level II, 16K. $14.95 for tape, $20.95 on disk.

FAST, REAL-TIME ACTION WITH SOUND



PINBALL

by John Allen

Get your flipper fingers ready for action in this real-time, machine language game.

Lots of sound and flashing graphics make this fast action game so much like the real thing that you'll have to remind yourself not to shake your TRS-80*. Choose from five playing speeds to match your skill—but be prepared tor alot of practice if you ever hope to master the fastest speed.

Can you beat your friends' scores? Will you avoid the dreaded "Bermuda Square?" Get PINBALL today and find out. Available for $14.95 on tape or $20.95 on disk.

*TRS-80 is s trademark of Tandy Corp.

Acorn 34

Software Products, Inc.

634 North Carolina Avenue, S.E., Washington, D.C. 20003

These and other popular Acorn programs are available now at fine computer stores. Ask for them.

DEALER INQUIRIES INVITED

Instant Software™

Now Sells The Adventure Series*

YOU CAN CALL 1-800-258-5473 TO ORDER THESE NEW INSTANT SOFTWARE OFFERINGS. OR STOP BY ONE OF OUR 300 DEALERS THROUGHOUT THE U.S.A. AND THE WORLD, FOR PERSONALIZED SERVICE.

*AND MORE

ADVENTURE!

ALL 3 PAKS
TRS-80 32K
Apple 48K

3 PAKS
Apple
No.5503AD
TRS-80
No.5504RD

1) **ADVENTURELAND** - You wander through an enchanted world trying to recover the 13 lost treasures. You'll encounter wild animals, magical beings, and many other perils and puzzles.
TRS-80 Tape Order No.5501R

2) **PIRATE'S ADVENTURE** - "Yo ho ho and a hottle of rum...."You'll meet up with the pirate and his daffy bird along with many strange sights as you attempt to go from your London flat to Treasure Island.
TRS-80 Tape Order No.5505R

3) **MISSION IMPOSSIBLE ADVENTURE** Good morning, your mission is to....and so it starts. Will you he able to complete your mission in time? Or is the world's first sutomated nucleas reactor doomed?
TRS-80 Tape Order No.5507R

3 PAKS
Apple
No.5509AD
TRS-80
No.5510RD

4) **VOODOO CASTLE** - Count Cristo has had a fiendish curse put on him hy his enemies. There he lies, with you his only hope.
TRS-80 Tape Order No.5508R

5) **TNE COUNT** - You wake up in a large hrass bed in a castle somewhere in Transylvania. Who are you, what are you doing bere, and WHY did the postman deliver a hottle of hlood?
TRS-80 Tape Order No.5511R

6) **STRANGE ODYSSEY** - Marooned at the edge of the galaxy, you've stumbled on the ruins of an ancient alien civilization complete with fabulous treasures and unearthly technologies.
TRS-80 Tape Order No.5512R

3 PAKS
Apple
No.5514AD
TRS-80
No.5515RD

7) **MYSTERY FUN HOUSE** - Can you find your way completely through the strangest Fun House in existence.
TRS-80 Tape Order No.5513R

8) **PYRAMID OF DOOM** - An Egyptian Treasure Hunt leads you into the dark recesses of a recently uncovered Pyramid.
TRS-80 Tape Order No.5516R

9) **GNOST TOWN** - Explore a deserted western mining town in search of 13 treasures from rattlesnakes to runaway borses, this Adventure's got them all!
TRS-80 Tape Order No.5517R

Prices: All Tapes 16K . . .$14.95
All 3 Paks Disk . . $39.95

MORE*

STAR TREK 3.5: Get those Klingon's! The newest, most sophisticated Star Trek version by the Grand Master - Lance Micklus. The top program of it's kind available.
TR5-80 Mod.1 L.II 16K
Order No.5518RD $19.95 Disk
Order No.5519R $14.95 Tape

SLAG: War gaming at it's best. Real time graphics combined with long range planning make this an exciting, fascinating game.
TRS-80 Mod.1 L.II 16K
Order No.5520R $14.95 Tape

ASTEROID: The real time, high resolution graphics game that's a smash hit at Arcades all over the world. Three levels of difficulty. Save your quarters.
Apple 2 Disk
Order No.5521AD $19.95

KID VENTURES: #1 Little Red Riding Hood. Allows your child to interact with the story, learning as they go. Designed for readers and non readers alike. Includes sound and play along cassette tape.
TRS-80 Mod. 1 16K
Order No.5522R $14.95 Tape to Disk

GALACTIC EMPIRE: Good strategy space war game. You as commander of Galactica's Imperial forces, must capture and hold the 20 inhabited worlds of the Galactic System.
TRS-80 Mod. 1 16K
Order No.5523R $14.95 Tape
Order No.5524RD $19.95 Disk

GALACTIC TRILOGY: Special all three games of the Trilogy - Galactic Empire, Galactic trader, and Galactic Revolution all on one disk at a special savings.
TRS-80 Mod. 1 16K
Order No.5525RD $39.95

★ **INTERACTIVE FICTION:** The computer sets the scene with a fictional situation. Then you become a character in the story. When its your turn to speak, you type in your response. The resulting dialogue and even the plot will depend on what you say.

SIX MICRO STORIES: An introduction to interactive fiction. Involves the reader in a variety of situations from heing a spy to a pilot in a doomed 747 and more.
Order No.5526RD $14.95 Disk TRS-80 Mod. 1 16K

LOCAL CALL FOR DEATH: A detective story considerably more cballenging them the above program.
Order No.5527RD $19.95 Disk TRS-80 Mod. 1 16K

TWO HEADS OF THE COIN: Psychological Mystery set in the London of Sherlock Holmes. Most challenging of all. Will tax your observational and imaginitive skills.
Order No.5528RD $19.95 Disk TRS-80 Mod. 1 16K

We Guarantee It!

Instant Software Guarantee

OUR PROGRAMS ARE GUARANTEED TO BE QUALITY PRODUCTS. IF NOT COMPLETELY SATISFIED YOU MAY RETURN THE PROGRAM WITHIN 60 DAYS. A CREDIT OR REPLACEMENT WILL BE WILLINGLY GIVEN FOR ANY REASON.

DEALERS:
Instant Software is offering you SUBSTANTIAL discounts when ordering these top selling programs. Just call toll-free 1-800-258-5473 to place your order. WE SHIP RIGHT AWAY!!! Call us, if you need any further information.

Instant Software™

PETERBOROUGH, NEW HAMPSHIRE 03458 603-924-7296

Speed up your I/O programming with this collection of subroutines and get a robot to boot!

COMPAC

Daniel M. Romanchik KB6NU
4178 Decoro #1
San Diego, CA 92122

This article is *not* about assembling robots. Not the way you think, anyway. It's all about how to use the assembly language capabilities of your TRS-80 more easily.

One of the more time-consuming jobs of assembly language programming is the I/O.

Below is a description of a collection of subroutines I call "COMPAC", which is short for "communications package".

COMPAC, an assembly of subroutines, displays a byte on the CRT, fetches a byte from the keyboard, displays a message, clears the CRT screen, adds spaces between characters, positions the cursor anywhere on the screen and puts back lines between blocks of text. These routines enable you to better use the keyboard and CRT in your assembly language programs.

At the end of this article is the program which tells about "assembling" a robot on the screen, once the groundwork is down.

Getting Going

CRT, to begin, is the subroutine which displays a byte on the CRT screen (see Listing). To use this, just load the ASCII code for the character you want into the A register, and call CRT. The first thing we do is save the contents of the DE and IY registers by pushing them onto the stack, because we use these registers in CRT. If we didn't, we'd lose the data in the registers.

The next instruction, CALL 33H, jumps to a subroutine in the Level II ROM (CALL works like the GOSUB in BASIC). The byte in the A register is displayed at the cursor, and the cursor incremented. Then, the program returns to our subroutine, and the original values of IY and DE are popped off the stack. The last instruction, RET, returns us to the main program.

The next subroutine, KBSCAN, scans the keyboard. It places the ASCII value of the next key into the A register. KBSCAN returns characters in ASCII, doesn't give numeric values, and returns one character at a time. It requires only one instruction: CALL KBSCAN. When KBSCAN is called, the contents of the DE and IY registers are saved as in CRT.

A Level II routine at 002BH is then called and does most of the work. It scans the keyboard once, and if any of the keys are pressed, places the ASCII value of the character into the A register. If none are pressed, it returns a value of 00H. When finished with the scan, execution is returned to KBSCAN.

The next two instructions check to see if any of the keys are pressed. CP 0 compares the value in the A register to 00H. If A = 00H (meaning no key was pressed), the Z flag is set to 1. If A<>00H (one is pressed), the Z flag is reset to 0.

The next instruction, JP Z,AGN, checks the condition of the Z flag, and jumps to the statement labeled AGN if the flag wes set. This sends us back to the scanning routine in the Level II ROM. We loop over and over until somebody hits a key. If the flag is reset (=0) we go to the next instruction, which pops the values of DE and IY off the stack. This routine can be used to stop program execution. It stays in a loop until a key is pressed.

The next subroutine, MESAGE, displays a message. The programmer must first store the message in memory and keep track of it. Say we wanted to display MY PROGRAM. Store the characters in the message at some known place in memory:

```
ORG   6000H
DEFM  'MY PROGRAM'
DEFB  0
```

We use 00H as our end-of-text character. To print this onto the CRT requires the following steps:

```
LD    IX,6000H
CALL  MESAGE
```

IX is located with the memory location of the first character and then the subroutine is called.

The next instruction gets a character from memory, and puts it in the A register. It is compared to 0. If the value is 0, we are at the end of the string. If not, there is a character to display, and we call the CRT routine to display it.

The next instruction, INC IX, increases the pointer, and JP AGAIN jumps to the beginning of the subroutine to fetch the next character. This time the IX register is pointing to the next character. We repeat these steps until we get to the "0" at the end of the string.

The CLRCRT subroutine is used to clear the screen. First, we load the A register with 1CH, and CALL CRT as if to display this byte. However, when the CRT subroutine sees 1CH, it resets the cursor to the home position. 1FH is then loaded into the A register and CRT is called. 1FH clears the screen from the current position of the cursor to the end of the screen. This subroutine requires one instruction—CALL CLRCRT.

Continue to 203

SAVE YOUR TIME ...AND YOUR MONEY.

Instant Software has the two best mail list programs available for your TRS-80 Model I and Model II.

Mail/List for Model I and Model II

This mailing list program maintains separate alphabetical and zip code files in *constant sort*. When you add a name to your list, it will *automatically* be inserted into its correct position in the files; therefore it's always ready to print labels!

It will record your information in these fields: NAME, ADDRESS, CITY, STATE, ZIP, PHONE NUMBER, PHONE EXTENSION and a five character CODE field. You have the choice of a 3 line, 4 line, or user defined label format. It can even include (optionally) a message line on your label!

The programs most outstanding feature is its sorting capabilities. Mail/List allows you to choose which names you want to be printed from the whole list. For example, all people in one zip code, or all people named Jones, who are living in a particular city or state. For any name in your list you can assign a code within the CODE field. You can then specify the code when printing labels, and only names with that code will be printed out. You can specify up to 9 different codes!

Every business and organization will save time and money with Mail/list to keep track of customers or members.

TRS-80 Model I version, Order No. 5000RD $99.00.
Requires 16K RAM, Expansion Interface with at least 16K RAM, one disk-drive and a printer.

TRS-80 Model II version, Order No.5001RD $199.00.
Requires 64K RAM and printer.

One-D Mailing List

Here is a mailing list system that can be run on only ONE disk-drive! You can have up to 17 fields of selection for name/address retrieval.

•Disk versatility allows you to add, delete, or change the numerous details stored in the system.

•Features of the One-D Mailing List includes:

•Automatic name sort (alphabetically or by ZIP code). •Rapid access to any name on file. •Easy error correction and recovery. •Prints selective name listings. •Revise or update listings at any time. •Up to 2500 names on-line (with 4 drives). •Prints a list of all names on file. •Prints mailing labels.

This package requires the following minimum system: 1. A TRS-80 Model I Level II microcomputer with 16K RAM. 2. An Expansion interface with 0 to 32K of RAM. 3. A single disk drive (extra drives optional). 4. A printer. 5. Any TRSDOS compatible Disk Operating System.

Order No. 0123RD $24.95.

WRITE FOR OUR NEW CATALOG

TO ORDER: See Your Local Instant Software Dealer or Call Toll-Free 1-800-258-5473.

Instant Software™

PETERBOROUGH, N.H. 03458

Merry Christmas From Your TRS-80*

TRS-80* 16K LEVEL II

COMPU CAROLS

FEATURING SOUND

ORDER NO. 0036R $9.95

Here is a selection of all time Christmas favorites. These beautiful Christmas Carols will be displayed verse-hy-verse, accompanied by clear, soft music on your TRS-80. The pretty music for each carol is played for your enjoyment over an AM radio, easily hooked up to your TRS-80. This lovely Christmas Album will bring hours of enjoyment from your Micro. The soft musical notes offer an ideal Christmas setting, seeming to blend perfectly with gently falling snow, as the flickering flames from the fireplace logs reflect from each glowing ornament on the tree. We wish you much enjoyment with Compucarols and a MERRY CHRISTMAS!

12 OF THE BEST LOVED SONGS OF CHRISTMAS

(1) AWAY IN A MANGER
(2) DECK THE HALLS
(3) GOD REST YE MERRY GENTLEMEN
(4) HARK! THE HERALD ANGELS SING
(5) JINGLE BELLS
(6) JOY TO THE WORLD
(7) NOEL
(8) O COME ALL YE FAITHFUL
(9) O LITTLE TOWN OF BETHLEHEM
(10) SILENT NIGHT
(11) WE THREE KINGS
(12) WHAT CHILD IS THIS?

TO ORDER:
1) See your local Instant Software Dealer
OR
2) Call toll-free 1-800-258-5473

Instant Software™

*A trademark of Tandy Corporation

PETERBOROUGH, N.H. 03458 603-924-7296

KING OF THE HILL!

We've taken artistic license with our illustration in order to make a point: MYCHESS is the most powerful microcomputer chess program on the market, bar none.

Proof? All you want and then some. For example, MYCHESS was the winner of the "Fifth West Coast Computer Fair". At the "Third World Computer Chess Championship" in Linz, Austria, it was the highest finishing micro... in addition to winning the special Blitz Tournament (5 to 1) against six top players. Add to this its USCF rating of 1565, and you know you're dealing with the King of the Hill.

You'll find MYCHESS is the perfect companion or opponent whether you're an advanced player, or starting your first game. For it lets you set the difficulty of the game from level 1 to 9. And, you can change levels of play as you go ... or even change sides. Want to set time limits for moves? MYCHESS can do it. Want to save a game for later? MYCHESS will store up to 6 games. And, for added interest, it will even predict the upcoming line of play.

If you're a player, you'll appreciate the MYCHESS challenge. If you're a beginner, you'll enjoy learning from a master. Either way, when it comes to superior chess, make your move ... to MYCHESS. Available for the TRS-80* with 32K, for $34.95 including disk, documentation and backing by Programma International. Apple** version coming soon.

Can **you** beat

MYCHESS

Surplus TRS-80* RAM Memory Chips

DUE TO CHANGES IN PRODUCT STRUCTURE AMERICAN BUSINESS COMPUTERS IS OFFERING SEVERAL THOUSAND 200 NANOSECOND RAM MEMORY CHIPS AT CLOSE-OUT PRICES.

$45 (per 16K set)

AMERICAN BUSINESS COMPUTERS GUARANTEES ALL MEMORY CHIPS TO BE BRAND NEW AND FREE FROM DEFECTS FOR 180 DAYS. PRICE ($45) INCLUDES 8 CHIPS. QUANTITY DISCOUNTS AVAILABLE.

AMERICAN BUSINESS COMPUTERS ✓397

*TM TANDY CORP.

118 S. MILL ST. - PRYOR, OK 74361 - 918-825-4844

!!!ANTS!!! A STIMULATING ACTION GAME $14.95

Two colonies of ants are at war. Opposing queen ants produce four types of offspring: workers, soldiers, guards, and drones. The challenging strategy is to produce ants in the proper sequence to sting the enemy queen or overrun it's nest.
SPECIAL FEATURES: Full screen action with SOUND. Hundreds of ants battle with machine language speed. 3 game variations. Play another person or the computer opponent at 4 difficulty levels.
EASY TO PLAY. CHALLENGING TO MASTER!

PARSECTOR V THE ULTIMATE SPACE WAR $19.95

Two opponents must navigate powerful mother ships through the galaxy and capture parsectors. Launch fleet battle craft: fylers, cruzers, and bases. Fire high powered energy beams or deadly short range weapon spreads. To win you must conquer the galaxy or destroy your opponent's mother ship.
SPECIAL FEATURES: Unique split screen gives each player a private video display. Action Sounds and Graphics: explosions, weapon releases, launches, and more. Variable galaxy size. Play another person or the computer opponent at 3 difficulty levels.
LAUNCH A COMPLETE SPACE FORCE IN MINUTES!

WITH SOUND

HIGH SPEED LIFE THE FASTEST $9.95

John H. Conway's famous game comes to life with full animation. The fastest, most advanced version. Many patterns average over 500 generations per minute. SPECIAL FEATURES: Full screen 48x64 matrix. 32 preprogrammed patterns. Special pattern generators. Create, edit & save your patterns on data tape. Sound. Single key functions: erase, save, single step, delay, scroll eight ways.
SPEEDS UP TO 2000 GENERATIONS PER MINUTE!

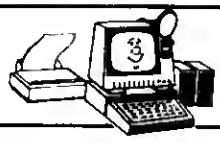
NAME THAT STATE QUIZ EDUCATIONAL $14.95

A fascinating way to learn about our 50 states. It teaches the state shapes, names, capitals, populations, areas, and geographic regions. Three types of quizzes: true & false, multiple choice, and fill in the blank. SPECIAL FEATURES: Action Sounds, rewarding tones and harassment buzzers. Continuous boots. Easy to use.

TRS 80 L2 16K FOR CASSETTE & INSTRUCTIONS
SEND CHECK OR MONEY ORDER TO:

Synergistic Solar Inc PO BOX 560595 MIAMI FL 33156

Please write for more information. Dealer inquiries invited.



✓214

THE MICRO CLINIC

CENTRONICS 779/RS PRINTER I LOWER CASE KIT

Don't let the newer low-priced printers with lower-case capabilities make your Centronics 779/Radio Shack Printer I obsolete. Our assembled and tested CLC-1 conversion kit will give your 779 the full upper/lower case character set at a fraction of the cost of a new printer. Illustrated instructions make installation easy - just 3 connections, no etch cuts. Compare our introductory price to other kits selling for $125 - at $99 our CLC-1 kit brings your 779 into the 80's and makes word processing a practical application.

CLC-1 INTRODUCTORY PRICE: $99 **Includes P/H CA add 6% tax.**

master charge — VISA/MC include card number, signature, exp. date, phone number. MC include interbank number. Introductory price good thru 1/31/81. — VISA

THE MICRO CLINIC • 17375 Brookhurst • Suite 114 • Fountain Valley, CA 92708

TRS 80

ROM

Wasn't built in a day!

And now need a good guide to explore it. SUPERMAP is an invaluable aid in discovering the hidden secrets of Level II ROM. SUPERMAP is a long detailed and commented memory map of ROM routines giving register setups for calling them. Reserved RAM and BOOT, SYS0, and SYS1 are commented. There are notes on the significant addresses in the Radio Shack Editor /Assembler, TBUG, and the Electric Pencil. Tape and memory format are given too, makes which version of Level II ROM you have and what the differences are. All this and more for only CSMOVE

We would be lost without SUPERMAP!

Fuller Software
630 E. Springdale
Grand Prairie, TX 75051
(214) 642-0441

✓102

VUTIL Examine, clear, initialize, move and modify data in memory. Compare two blocks or search for up to 32 byte HEX or ASCII string. Punch, load, verify, or execute SYSTEM *apes. Set breakpoints and jump. Display and modify registers. Output to printer if desired. TVUTIL tape has a bonus program, MODIFY improves EDTASM(1.1 or 1.2) by allowing return to BASIC without loss of source file. Also uses Level II I/O to allow debounce, serial drivers, etc. (16-48) $15
DUTIL Has all the above plus disk read and write. (16-48K DOS) $20

DISK-MOD modify EDTASM(1.1 or 1.2) under TRSDOS, NEWDOS, even VTOS into a disk type E/A. Adds block move, global change, paged output to printer with optional page prompt, sorted symbol table, and OBJN conversions. Protect memory, use Level II I/O block; recover from reboots, and branch to any address. DIR, FREE, and KILL available without leaving the modified E/A (32K+ DOS required). $20

DISABLE I Disassemble ROM code to video, printer or tape. Provides XQUotes ORG, and Sven Imedia. Tapes load into E/A to allow you to modify machine code programs easily. 16K, 32K, and 48K! $15
DISABLE II DOS version writes E/A or MACRO disk file. (32K DOS) $20

STEPRO still single step BASIC or machine language programs even in ROM. Outputs trace values to video or printer via DCB. STEP80 in ROM. Outputs trace values to video or printer via DCB. STEP80 more. Relocatable progrm. $16.95 + $1.00 postage and handling.

SILVER-IT a kit to silver solder the TRS80's cheap bus connections for increased reliability and ease. Hi-Rs silver wire for only $5.

INSIDE LEVEL II excellent manual containing 18 chapters on using ROM routines with many examples and explanations. It shows how to make a composite program load mode; SYSTEM but execute in both BASIC and machine language. Problems of using ROM in a disk system are shown and solved. Expand USR; relocate and interface with BASIC programs after reading these chapters. $15.95 + $1.00 postage and handling.

THE BOOK The most comprehensive and complete book yet on the math routines in Level II ROM. Has a commented disassembly of addresses 704 to 1167. This is the BOOK for the serious assembly programmer. 138 pages. 7 chapters. $18.95 plus $1.50 postage and handling.

DOS Bytes BOOT, SYS0/SYS, SYS1/SYS comments. 12 page booklet. $5

THE ALTERNATE SOURCE

The magazine of advanced applications and software for the TRS-80

✓138

Presents:

TRAKCESS!

by: Roxton Baker

- *Most powerful disk utility yet!*
- *"Smart" DUPLICATE command analyzes every track! (2 drives)*
- *Super Pencil-type Editing!*
- *Read/Write/Edit/Create—any—track or sector, standard or not!*
- *A great tool for learning about disk structures—including the formerly little documented info!*
- *Fingertip control of all disk functions on 1-4 drives.*
- *Recover damaged disks. Oddball tracks and sectors don't faze it. Feed it TRSDOS, CP/M, protected disks, almost anything!*
- *Use it to create special disks interactively!*

An experiment!

An experiment to make this incredible program available to all: Add 1 cent for each copy (up to 5) you want to make and resell for your friends. We license you. The idea is to make TRAKCESS available to everyone who needs it. Appreciate this effort? Make it a success—Order now! $24.95 on Disk. Trakcess is over 23K of code. It requires 48K to run. Handles any number of tracks.

MC/VISA/COD—Add $1.25

Phone: 517/485-0344

TAS, 1806 Ada, Lansing, MI 48910

TRS-80 Owners: Turn Your Typewriter Into A Printer

...With the **KGS~80** Keyboard Actuator

- Plug-in compatibility with the TRS-80 . . . attractive enclosure contains actuator and interface.
- Least expensive way to get letter quality printing.
- No mechanical modifications to the typewriter are necessary.
- Rests firmly above the typewriter keyboard. Can be installed or removed in 5 seconds.
- *Does not require any software to operate . . .* works with Pencil, Scripsit and other word processing programs.
- Solenoids with soft plastic tips strike typewriter keys with the same force a typist would exert.

✓149

Dealer Inquiries Invited.

KGS KOGYOSHA CO., LTD.
179 Riveredge Rd., Tenafly, N.J. 07670 (201) 569-8769

SPACES lets you put up to 255 spaces between characters. Do it this way. Load the B register with the number of spaces you want and call SPACES, as in the following:

```
LD     B,10
CALL   SPACES
```

The A register is loaded with 20H, which is the ASCII code for space. Call CRT which displays the space. The next instruction, DJNZ SPACES, does this:

- Decrements the B register.
- Checks if B equals 0.
- If B is not equal to 0, it jumps to the statement labelled SPACES.
- If B equals 0, it continues.

It causes a number of loops, equal to the initial value of the B register, to be executed.

The next subroutine, POSIT, positions the cursor. Load the BC registers with the desired positions, and call POSIT. Try this:

```
LD     BC,512
CALL   POSIT
```

to place the cursor at position 512. The contents of the HL registers are saved, as in CRT. HL is loaded with the value 3C00H which is the start of the video RAM, or position 0. Next, add BC to HL. Load it into memory location 4020H. 4020H and 4021H contain the cursor position. Restore HL to its original value by popping its contents off the stack. Return to the main program.

Last Step

LINES is the last subroutine and places blank lines between text. To get started, load the B register with the number of lines to be inserted. For example, to place two lines between messages, do this:

```
LD     B,2
CALL   LINES
```

Program Listing

```
                00090 ;*** BEGINNING OF COMPAC ***
7C00            00100       ORG     7C00H    ;LOAD COMPAC AT LOCATION 7C00H
                00104 ;CRT IS THE BYTE DISPLAY ROUTINE
                00105 ;A REG MUST CONTAIN BYTE TO BE DISPLAYED
7C00 D5         00110 CRT   PUSH    DE
7C01 FDE5       00120       PUSH    IY
7C03 CD3300     00130       CALL    33H      ;33H IS ENTRY POINT FOR
7C06 FDE1       00140       POP     IY
7C08 D1         00150       POP     DE
7C09 C9         00160       RET
                00165 ;KBSCAN GETS A CHARACTER FROM THE KEYBOARD
7C0A D5         00170 KBSCAN  PUSH    DE
7C0B FDE5       00180       PUSH    IY
7C0D CD2B00     00190 AGN   CALL    2BH      ;2BH IS ENTRY POINT FOR
7C10 B7         00200       OR      A        ;IF NO KEY PRESSED,00H RETURNED
7C11 28FA       00210       JR      Z,AGN    ;IF BYTE=00H,SCAN AGAIN
7C13 FDE1       00220       POP     IY
7C15 D1         00230       POP     DE
7C16 C9         00240       RET              ;BYTE WILL BE IN A REG
                00245 ;MESAGE DISPLAYS A MESSAGE ON SCREEN
7C17 DD7E00     00250 MESAGE  LD      A,(IX)
7C1A FE00       00260       CP      0        ;CHECK FOR END OF STRING
7C1C CA277C     00270       JP      Z,RETN   ;IF END OF STRING, RETURN
7C1F CD007C     00280       CALL    CRT      ;DISPLAY BYTE
7C22 DD23       00290       INC     IX       ;INCREMENT POINTER
7C24 C3177C     00300       JP      MESAGE   ;GO BACK FOR ANOTHER BYTE
7C27 C9         00310 RETN  RET
                00315 ;CLRCRT BLANKS THE SCREEN
7C28 3E1C       00320 CLRCRT  LD      A,1CH    ;1CH IS SPECIAL CHARACTER -
7C2A CD007C     00330       CALL    CRT
7C2D 3E1F       00340       LD      A,1FH    ;1FH IS SPECIAL CHARACTER -
7C2F CD007C     00350       CALL    CRT
7C32 C9         00360       RET
                00365 ;SPACES INSERTS A NUMBER OF SPACES
7C33 3E20       00370 SPACES  LD      A,20H    ;20H IS ASCII CHAR FOR SPACE
7C35 CD007C     00380       CALL    CRT      ;B REG CONTAINS # OF SPACES
7C38 10F9       00390       DJNZ    SPACES
7C3A C9         00400       RET
                00405 ;POSIT POSITIONS CURSOR
                00406 ;BC CONTAINS THE CURSOR POSITION DESIRED
7C3B E5         00410 POSIT   PUSH    HL
7C3C 21003C     00420       LD      HL,3C00H
7C3F 09         00430       ADD     HL,BC
```

Program continues

```
7C40 222040    00440          LD      (4020H),HL
7C43 E1        00450          POP     HL
7C44 C9        00460          RET
               00465 ;LINES INSERTS A NUMBER OF BLANK LINES
               00466 ;BC CONTAINS THE NUMBER DESIRED
7C45 3E13      00470 LINES    LD      A,13H   ;13H IS ASCII FOR NEWLINE
7C47 CD007C    00480 NXTLIN   CALL    CRT
7C4A 10FB      00490          DJNZ    NXTLIN
7C4C C9        00500          RET
               00510
               00520 ;*** END OF CONPAC ***
               00530
               00560 ;LOAD MESSAGE INTO MEMORY AT 7A00H
7A00           00570          ORG     7A00H


7A00 41        00580          DEFM    'A ROBOT'
7A07 00        00590          DEFB    0
               00600
               00610 ;DISPLA IS USED TO DISPLA A FULL BLOCK AT THE CURSOR
7A08 3EBF      00620 DISPLA   LD      A,0BFH  ;0BFH IS ASCII FOR THE FULL
7A0A CD007C    00630          CALL    CRT     ;BLOCK GRAPHICS CHARACTER
7A0D C9        00640          RET
               00650
               00660 ;BODY DISPLAYS ONE BODY SEGMENT
7A0E CD007A    00670 BODY     CALL    DISPLA
7A11 0601      00680          LD      B,1
7A13 CD337C    00690          CALL    SPACES
7A16 0603      00700          LD      B,3
7A18 CD007A    00710 TRUNK    CALL    DISPLA
7A1B 10FB      00720          DJNZ    TRUNK
7A1D 0601      00730          LD      B,1
7A1F CD337C    00740          CALL    SPACES
7A22 CD007A    00750          CALL    DISPLA
7A25 C9        00760          RET
               00770
               00780 ;LEGS DISPLAYS ONE LEGS SEGMENT
7A26 CD007A    00790 LEGS     CALL    DISPLA
7A29 0601      00800          LD      B,1
7A2B CD337C    00810          CALL    SPACES
7A2E CD007A    00820          CALL    DISPLA
7A31 C9        00830          RET
               00840
               00850 ;BEGINNING OF MAIN PROGRAM,STARTS AT 7000H
7000           00860          ORG     7000H
7000 CD287C    00870 BEGIN    CALL CLRCRT     ;CLEAR SCREEN
7003 0602      00880          LD      B,2     ;SKIP TWO LINES
7005 CD457C    00890          CALL    LINES
               00900
7008 061C      00910 MSG      LD      B,28    ;INSERT 28 SPACES
700A CD337C    00920          CALL    SPACES  ;TO CENTER MESSAGE
700D 0D21007A  00930          LD      IX,7A00H;LOAD POINTER
7011 CD177C    00940          CALL    MESAGE  ;DISPLAY MESSAGE
               00950
7014 011F02    00960          LD      BC,543  ;POS CURSOR AT 543
7017 CD387C    00970          CALL    POSIT
701A 0603      00980          LD      B,3     ;DISPLAY 3 BLOCKS FOR HEAD
701C CD007A    00990 HEAD     CALL    DISPLA
701F 10FB      01000          DJNZ    HEAD
               01010
7021 015D02    01020          LD      BC,605  ;POS CURSOR AT 605
7024 CD3B7C    01030          CALL    POSIT
7027 0607      01040          LD      B,7     ;DISPLAY 7 BLOCKS FOR SHOULDERS
7029 CD007A    01050 SHOLDR   CALL    DISPLA
702C 10FB      01060          DJNZ    SHOLDR
               01070
702E 019D02    01080          LD      BC,669  ;POS CURSOR AT 669
7031 CD387C    01090          CALL    POSIT
7034 CD0E7A    01100          CALL    BODY    ;DISPLA ONE BODY SEGMENT
               01110
7037 01DD02    01120          LD      BC,733  ;POS CURSOR AT 733
703A CD387C    01130          CALL    POSIT
703D CD0E7A    01140          CALL    BODY    ;DISPLA 2ND BODY SEGMENT
               01150
7040 011F03    01160          LD      BC,799  ;POS CURSOR AT 799
7043 CD387C    01170          CALL    POSIT
7046 CD267A    01180          CALL    LEGS    ;DISPLA ONE LEGS SEGMENT
               01190
7049 015F03    01200          LD      BC,863  ;POS CURSOR AT 863
704C CD387C    01210          CALL    POSIT
704F CD267A    01220          CALL    LEGS    ;DISPLA 2ND LEGS SEGMENT
               01230
7052 CD0A7C    01240          CALL    KBSCAN  ;PROGRAM STOPS TO ALLOW YOU TO
               01250                          ; SEE PICTURE UNTIL YOU HIT A KEY
7055 CD287C    01260          CALL    CLRCRT  ;CLEARS SCREEN
               01270 ;
7058 C3191A    01280          JP      1A19H   ;RETURN TO BASIC
               01290 ;
7000           01300          END     7000H
```

The A register is loaded with 13H, the ASCII code for a new line, and the CRT subroutine is called. This causes the cursor to be positioned at the beginning of the next line. The next instruction, DJNZ NXTLIN, decreases the B register and repeats the instruction until the register is zero.

Robots

To illustrate the subroutines, write a simple program (see Listing).

Now we come to the robots:

- Skip two lines.
- Print the title, "A ROBOT".
- Print a picture of a robot using the TRS-80 graphics.

The first thing we do is load the message into memory. ORG 7A00H defines the place in memory and DEFM defines the characters we want. DEF 0 places a 0 in the next location. DISPLA is the first routine used to display one block. 191 is the value of its ASCII code.

The next two routines, BODY and LEGS, display one body and one leg segment. A body segment consists of a block, a space, three blocks, a space and another block. The leg is made of a block, a space and another block.

The first statement of the main program, ORG 7000H, sets the place in memory where the program will reside. Clear the screen. Skip two lines by loading the B register with 2 and calling the LINES subroutine. To display the title, display 28 spaces in order to center the message. Load the IX register with the starting address of the message and call the MESAGE subroutine to display A ROBOT. The next part is the actual display of the robot's image! These statements position the cursor and display the appropriate parts of the robot.

Then we call KBSCAN to stop execution, and admire the picture we just displayed.

The robot will stay until we hit a key. After this we clear the screen again.

The last statement, JP 1A19H, returns us to BASIC.■

THE AFFORDABLE HOME COMPUTER

IS NOW ON SALE.

When PMC-80 was first introduced to the United States, the response was overwhelming! The Computer World was ASTONISHED at the QUALITY, as well as the PRICE. In fact, the PMC-80 has almost all the features of America's best selling computer, the TRS-80, but with a price tag of $200.00 less! (SIMUTEK'S price is $275.00 less!)

Microsoft's Level II Basic and 16K Memory.

Another reason for all the commotion is that the PMC-80 uses the same, easy to learn, LEVEL II BASIC language that the TRS-80 uses! What does this mean? It means that the PMC-80 can run all the 1000's of programs that have been written for the TRS-80 Level II, 16K computer! Some of the programs available include: Flight simulation, World Champion Chess program, Scores of educational and business programs. Word processing programs and hundreds of other games and simulations.

The PMC-80 is expandable!

Your PMC-80 is ready to grow with your needs. Using a special cable, available from Simutek for $35.00, it may be connected to Radio Shack's Expansion interface, to give you up to 48,000 characters of memory, up to 4 disk drives, addition of a telephone communication system, Voice Synthesizer, various printers, a real time clock, as well as plotters and other neat interfaces! As your skills with the PMC-80 improve, you're sure to want some of the ADD-ON's described above. (And these are just a few!)

Save Money! Use your own television!

The PMC-80 has a built in RF MODULATOR so you can use your black and white or color TV for a VIDEO MONITOR! A simple hook-up to your television's antenna connector, makes channel 3 your computer's video channel.

Special Introductory Offer: 25 Free Programs

SIMUTEK, a leading innovator in Home Computer Software, is making a SPECTACULAR INTRODUCTORY OFFER to people that ORDER the PMC-80 NOW. With each purchase, we will give 25 FREE HOME COMPUTER PROGRAMS! Some of these include: Home Amortization tables program, Loan payment programs, Depreciation rate program, Interest table program, Annuity and Investment calculation programs as well as these great animated games: GRAPHIC-TREK 2000: Command the Enterprise!, INVASION WORG . Stop the invading marauders from space before they take over earth! You command Earth's forces of androids, space fighters, laser guns etc., against the enemy's robots, saucers, proton destroyers, etc!, STAR WARS: Fly your space fighter into the Death Star to destroy it! But watch out, Darth Vader doesn't like you!, SPACE TARGET: A fantastic animated arcade game of skill and daring!, SAUCERS: Can you win win the coveted Medal of Honor?

Comparison Chart

Features	PMC-80	TRS-80
Microsoft's Fantastic Level II Basic	Yes	Yes
Full 128 x 48 Graphics	Yes	Yes
16,000 characters memory	Yes	Yes
Tape recorder for storing or retrieving programs	Yes	Yes
Use your own TV (Save $$)	Yes	No
Expandable to 48,000 characters of in computer memory	Yes	Yes
Use TRS-80 expansion interface	Yes	Yes
Expandable to 4 floppy disk drives (over 100,000 characters of storage on each one!)	Yes	Yes
Telephone Communications available connect to large computers/electronic mail etc	Yes	Yes
1000's of ready made programs avarible for "educational" and "scientific" applications?	Yes	Yes
Printers available	Yes	Yes
High Speed Z80 CPU	Yes	Yes
Interface available for controlling lights and appliances in home	Yes	Yes
Retail Price	**$646.00**	**$649.00**

Here's what you get:

The PMC-80 microcomputer with 16,000 characters of "In Computer Memory", Microsoft's Level II Basic (built into the computer), a cassette player for storing or retrieving programs or data (cassette player is built into the computer!), an RF Modulator for connecting the PMC-80 to your television set, 25 FREE programs so you start using your computer immediately, complete instruction manual, learning manual and owners manual so you can begin writing your own programs right away!

Best of all, you have the chance to use the PMC-80 in your own home before making your final commitment! Keep it for two weeks, if, for any reason you decide not to become a PMC-80 owner simply send it back, (in new condition please), and we will promptly refund the full amount, including your delivery charge!

Time is of the essence. Please order now, as this price can only be guaranteed through December 25, 1980.

Order Now Save $76.00

Credit card holders may use our TOLL FREE NUMBER. Or send check for $569.00 plus $6.00 delivery. (Arizona residents add $23.80 state tax). Please mention this magazine. (No tax on out of state orders).

Call Toll Free
800-528-1149
In Arizona call 602-886-5880 Collect

SIMUTEK COMPUTER PRODUCTS™ ✓19
9881 E. Skyview Drive
Tucson, AZ 85710

TRS-80 is a registered trademark of Radio Shack, a Tandy Corp

Wow! Automatic activation for the Shack's line printer, with software control, no less.

Turn-On

Dr. J. H. Nestor
39114 Rt. 303
Grafton, OH 44044

After using the Radio Shack Tractor Feed Line Printer with my TRS-80 for about two months, I came to two conclusions: It is a reasonable machine for personal and small business use; but something had to be done about that damned on/off switch!

Centronics, who manufactures the printer as their model 779, saw fit, for some perverted reason, to employ both a power switch and a print switch. The print switch controls the printer electronics, and is conveniently located on the front panel with an LED indicator.

Supposedly, this switch is turned on only when actually printing. However, since there is little current drawn, most users leave this switch in the on position all the time.

The power switch, which controls the motors, cooling fan, and AC to the printer, is located on the back panel, where it is difficult to reach.

No Noisier than Most

So, why not just turn on the printer when you bring up the computer? Those who would ask this type of question have never heard this printer in operation. Actually, it is no noisier than most while it's printing.

But try to concentrate on your programming with the machine idling. The constant whirring, buzzing, growling and slapping of drive belts is guaranteed to drive you to distraction in minutes.

Now you quip, "Big Deal! Just reach over and turn it off!"

Dear friend, unless your arms are seven feet long, you can't reach that switch. Even if the printer is sitting on the same table with the TRS-80 you still can't reach the switch without getting out of the chair and leaning over the back. Too much like work!

Let me digress for a moment to touch on my philosophy of microcomputing. For years, I have watched with fascination as TV

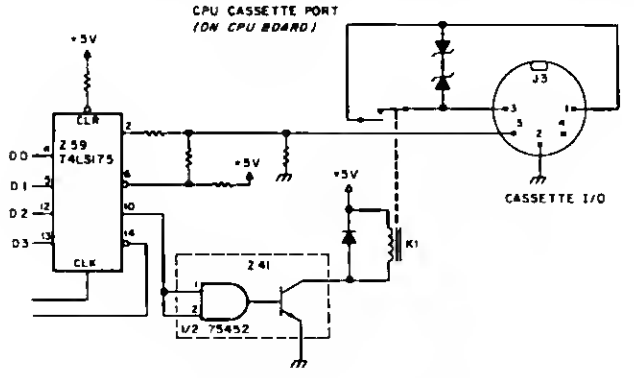


Fig. 1. Cassette Port on CPU Board

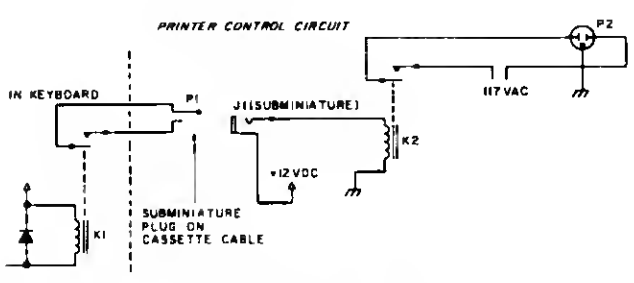


Fig. 2. Printer Control Circuit

```
10 CLS:PRINT
20 PRINT"DEMONSTRATION OF PRINTER CONTROL"
30 PRINT
40 INPUT"WHICH CASSETTE PORT ARE YOU USING (1 OR 2)";A
50 IF A=2 THEN POKE 14300,1
60 PRINT:PRINT"THE PRINTER WILL START.....NOW"
70 OUT 255,4:FOR X=1 TO 200:NEXT
80 LPRINT"THIS IS A DEMONSTRATION OF TRS-80 SOFTWARE FO
   R"
90 LPRINT"ON/OFF CONTROL OF THE LINE PRINTER."
100 LPRINT"THE PRINTER WILL STOP & THERE WILL BE A DELA
    Y"
110 LPRINT"STARTING.....NOW"
120 OUT 255,0
194 PRINT"THERE WILL BE A DELAY...PLEASE WAIT"
140 FOR X=1 TO 5000
150 NEXT X
160 PRINT:PRINT"NOW RESUME PRINTING....."
170 OUT 255,4:FOR X=1 TO 200:NEXT
180 LPRINT"THE PRINTER IS ON ONCE AGAIN. THIS CONCLUDES
    "
190 LPRINT"THE DEMONSTRATION. YOU SHOULD BE ABLE TO SEE
    HOW"
200 LPRINT"THE COMMANDS CAN BE INCLUDED IN ANY BASIC PR
    OGRAM."
210 OUT 255,0
220 PRINT:PRINT"END OF PRINTER ON/OFF CONTROL DEMONSTRA
    TION"
230 END


DONE
```

Program Listing 1. Demonstration Program

Goodies from GALACTIC

Specialty Programs for

TRS-80 Model I - II - III

EDAS 4.0 (Editor/Assembler)

This is the highly acclaimed *"USER ORIENTED"* Assembler for the TRS-80 Model II by GALACTIC. Loaded with features such as assemble to memory, block move, link to debugger, default filenaming, reverse video editing, warm start entry and much more. Now the programmer can write, assemble, test, and debug his code without ever leaving EDAS.

EDAS 4.0 with complete manual (120 pages)

Model II Version .... Was $229.00

NOW ONLY $179.00

MASS/MAIL SYSTEM

This is the NAME and ADDRESS system for subscription control or large mailing lists. It will handle up to 10,500 records, with a worst access time of less than 15 seconds and usual access of less than one second. All adds, deletes, and edits are instant for the operator and are then completed later in a "batch monitor". Extensive documentation and ongoing support. Requires TRS-80 Model II and 2 disk drives minimum. Contact GALACTIC direct for detailed specifications and prices for your exact needs.

Model II Version .... Contact GALACTIC for Price

STOCK MARKET MONITOR

This day to day market monitor is designed for the active trader. The system will track the performance of an issue against the market as well as against itself. The package comes with complete documentation and explainations of the formulas that are used by the program. The system is available for the Model I and the Model III TRS-80

Model I and III cassette version $89.00

Model I and III disk version $99.00

INVENTORY MASTER

Tired of being a slave to an out-of-control inventory? Let GALACTIC'S INVENTORY MASTER put you in control of your inventory. INVENTORY MASTER operates on a TRS-80 Model I and Model III 48K disk system (Minimum of 2 drives with capabilities of up to 4 drives) Drive spanning capabilities allow you to track 2700 inventory items with a 4 drive system (5100 items for the Model III). Unique machine language sort allows for instantaneous item insertion (approx. 15 seconds with 2700 items in system). Item access can be immediate using system-supplied control numbers. Modeled after a proven main-frame system costing tens of thousands of dollars. Complete add/edit/delete capabilities supported. Placement of orders can be machine-generated as well as user-generated, with editing capabilities. Full report-generator included. Exquisitely documented.

Model I Version $159.00

Model III Version $259.00

MODEL II HOST I/O SYSTEM

From the original author of the TRS-80 HOST and TERM systems in the RADIO SHACK "COMMUNICATIONS PACKAGE". This system allows the full control of the HOST facility by your BASIC program. Set the number of nulls to be sent after a C/R, set a command line to be executed if carrier is lost, turn HOST on and off, switch to channel A or B as desired, enable and disable the ability for the remote terminal to "BREAK" BASIC, identify whether a character came from the HOST'S keyboard or from the REMOTE'S and more. No knowledge of assembler needed. All options may be accessed from BASIC or ASSEMBLER. Complete with detailed documentation. Don't isolate your Model II, Let outside terminals access it's computing power.

Model II with TRSDOS 1.2 .... $179.00

Model II with TRSDOS 2.0 .... $199.00

MAIL/FILE SYSTEM

This is the name, address, phone number data base manager that has set the standard by which other systems are compared. This system contains advanced editing and output capabilities. The TRS-80 Model I system will handle up to 600 records per file, while the Model III version will handle up to 1150 records and the Model II will handle 2500 records per file. All versions are file compatable and maintain constant sort indexes on both NAME and ZIP CODE. International PHONE numbers and ZIP CODES are supported. Thousands of code combinations are available. The Model II version also has a "word processor" type input editor and fast assembler sorting. Complete documentation is included with each version of MAIL/FILE.

Model I Version .... $ 99.00

Model III Version $149.00

Model II Version $199.00

ULTRA TREK

This is an all new concept for this type of game, and compares to the others like chess compares to checkers. ULTRA-TREK is a complex, logical game, intended for the serious contestant. It is doubtful that you will ever master this game, but you will certainly enjoy trying! This program requires a TRS-80 Level II, 16K or more. The program is written totally in BASIC and uses 15.5K of RAM.

Model I & Model III Version
(cassette only) $14.95

galactic software ltd. ✓254

A Division of GS & WS, Inc

11520 N. Port Washington Rd.
Mequon, Wisconsin 53092
(414) 241-8030

Money Orders & COD's Shipped Within 24 Hours. Checks allow 2 weeks

---

INTEGRATED Disk KEYPLUS UTILITY PACKAGE

Disk Keyplus is a powerful multi-purpose utility for your TRS-80. Designed for ease of use, routines can be enabled or disabled in just two key strokes.

Disk Keyplus supports auto-repeat, lowercase video software, restoration of lost BASIC programs, single key stroke user definable strings, BASIC shorthand, direct graphic character input, lowercase without shift and more!

Disk based utilities include a routine that generates an user defined string at power up or at the stroke of two keys. More flexible than the DOS AUTO command, Disk Keyplus will execute any combination of commands or programs automatically.

Another routine allows users to initialize Disk Keyplus with any combination of utilities enabled or disabled.

Disk Keyplus is compatible with either TRSDOS or NEWDOS. A cassette with both the 32K and 48K versions is available for only $19.95. Non-disk Keyplus (Lv. 2, 16K) without the disk based utilities, but with keyboard debounce, loads in just 20 seconds and is available for $14.95. Pa. residents add 6% sales tax.

SJW, INC.
P.O. BOX 438
HUNTINGDON VALLEY,
PA 19006
215-947-2057

✓244

TRS-80 is a registered trademark of TANDY CORP.

Have computer, will travel.

Executive Computer System Carrying Cases.

- Makes your microcomputer truly portable.
- Protects your equipment: locking latches limit access.
- Rugged black vinyl with metal corners outside.
- Protective foam rubber, black velveteen covered, inside.
- Computer can be operated without removing from case.
- And cases are custom designed for full systems.

Apple* Executive Case holds:

- Apple microcomputer.
- 9" Sanyo monitor.
- 2 disk drives.
- Power strip.
- 2 boxes diskettes.
- Manuals.
- Dimensions: 28" x 21" x 10½"
- Weight: 17 pounds.
- Price: $179

TRS-80** Executive Case holds:

- TRS-80 Microcomputer
- Expansion interface.
- 2 disk drives.
- Power strip.
- 2 boxes diskettes.
- Manuals.
- Dimensions: 28" x 21½" x 8½"
- Weight: 17 pounds.
- Price: $179

Terms: FOB Los Angeles—Master Charge, Visa or check with order. Allow 3-4 weeks for delivery.

*Registered, Apple Computers, Inc.
**Registered Trademark, Tandy Corporation.

COMPUTER TEXTile ✓390

10960 Wilshire Blvd. Suite 1504
Los Angeles, CA 90024
(213) 477-2196

J1	subminiature phone jack Radio Shack #274-251
K2	OPOT Relay Radio Shack #275-206
P2	AC sockat grounded type
Misc.	3 conductor line card & plug mini-box, etc.

Parts List

and movies have shown how computers are used in their world. "Turn on the vectored whatsits." "Zoom in on that monster." "Disarm those missiles." "Blast 'em!"

Amazing! Those guys are controlling motors, reading meters, and launching missiles from that terminal. They don't have to get up and turn on any printers, so, why should I?

To achieve my goal of complete keyboard control, I have spent countless hours soldering, drilling and stringing wires so that, now, I don't have to leave the keyboard for anything but a beer (and I'm working on that).

My TRS-80 dials the phone, turns on the lights and sends Morse code on my amateur radio rig—all without any extra switches. Now, I can even control my printer, and I've decided to share the secret with you.

Relay Control

My approach is simple. I leave the power and print switches on and control the AC to the printer with a relay. The TRS-80 controls the relay via software commands.

The first secret lies in the use of the cassette relay as a control. The cassette motor is controlled by a small relay which is driven by a latched output port in the CPU keyboard unit (Fig. 1).

The relay is small, so don't get any ideas about switching heavy loads with it. But, you can use it to control another heavier load relay. That is how my system works.

Fig. 2 is a schematic of my circuit, using a Radio Shack relay #275-206. It is rated to switch three amps at 120 volts AC; the coil is 12 volts DC at 50 milliamps. This load is well within the limits of the relay in the TRS-80.

This isn't a construction article, as such, so I'll leave the details to you. Wiring is strictly non-critical. I stole 12 volts of DC from another source, but you could add a small DC supply for the relay. A different relay can, of course, be substituted so long as it will handle about two and a half amps, and the coil doesn't draw more than one-half amp. Of course, the less current through the CPU relay the better. My relay is housed in a cabinet but this, too, is non-critical.

If you have an expansion interface, there is another relay in that unit. It is a 4POT switch used to select either of two cassette machines. The CPU relay turns on the cassette motor, while the interface relay decides which cassette port will be activated. Fig. 3 shows how this system operates.

```
10 CLS
20 PRINT
30 PRINT"PRINTER STATUS CHECK"
40 B=PEEK(14312)
50 PRINT
60 PRINT"THE VALUE AT ADDRESS 14312 IS ";B
70 IF B<>127 THEN 120
75 PRINT
80 PRINT"THE PRINTER IS OUT OF PAPER"
90 PRINT"INSERT SOME MORE PAPER, THEN PRESS ENTER";
100 INPUT A$
110 GOTO 10
120 PRINT:PRINT"THERE IS PAPER IN THE PRINTER..."
130 IF B>127 THEN PRINT"BUT THERE IS STILL SOMETHING WR
    ONG":       PRINT "SHUT DOWN THE SYSTEM AND CHECK
     THINGS OUT";END
140 IF B<127 THEN PRINT"AND ALL SYSTEMS ARE READY TO PR
    INT"
150 END


DONE
```

Program Listing 2. Printer Status Check

If you are not using a cassette recorder with your TRS-80, you can connect your relay to cassette port 1.

If you still use a cassette machine occasionally, you can connect the cassette to port 1 and the printer relay to port 2.

Software Commands

I hate to disappoint some of you, but the software is also simple. The commands OUT 255,4 turn on the printer and OUT 255,0 turn it off.

These commands can be included as part of a BASIC program. OUT 255,4 is issued just before the first LPRINT statement, and OUT 255,0 is used after the last LPRINT operation.

The OUT 255,4 statement latches the relay on until the program ends and READY appears, or until an OUT 255,0 is executed.

If you wish to control the printer from cassette port 2, an additional bit of software is needed. Memory location 14308 decimal contains the cassette port selection code. On power-up this value is set to 0. This activates cassette port 1. A POKE 14308,1 command tranfers control to port 2.

You can perform this step at the beginning of a program or any time before the first LPRINT statement. It is only needed once in the program. However, if the program uses the cassette system for any other purpose, a POKE 14308,0 must be used to restore normal cassette operation. This includes CSAVE, CLOAD, PRINT #, and INPUT # statements.

Both the POKE 14308,(1 or 0) and the OUT 255,(4 or 0) can also be issued in the command mode. For example, if you want to LLIST a program, typing POKE 14308,1:OUT 255,4 and pressing the ENTER key will turn on the printer and print the program listing. OUT 255,0 is not needed, since the system returns to READY after the list is printed. Of course, if your relay is connected to cassette port 1, the POKE 14308,1 is not required.

If you want to get fancy, some additional software features are available to you. When the printer electronics are turned on a status signal is sent to the printer port in the expansion interface. It indicates whether or not the printer is turned on, and if it is out of paper. This printer status value is found in memory location 14312 decimal.

By PEEKing into that address, we can determine if the printer is ready to run. A value of 127 indicates that the printer is out of paper. If this is the case, a message can be printed on the screen, and the program interrupted.

A value of <127 signals that the printer is ready to run. You may want to play around with different uses for this printer status information. I'll leave it up to you. ■

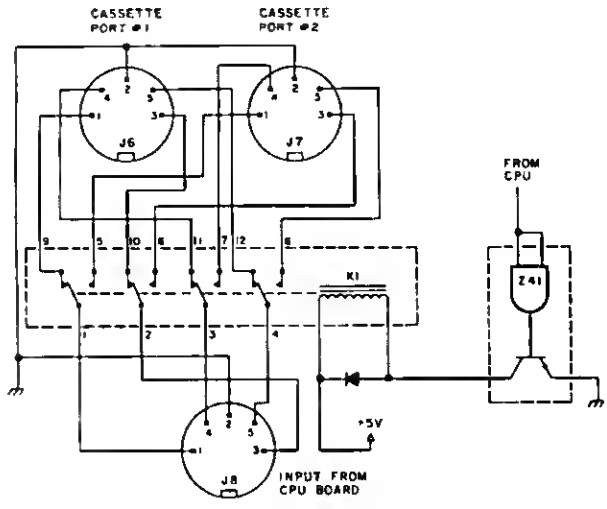


Fig. 3. Expansion Interface Cassette Ports

Using EDTASM to enhance itself.

Assemble It Yourself

Richard Koch
2740 Washington St.
Eugene, OR 97405

Radio Shack sells an excellent editor/assembler for the TRS-80. Unfortunately, this assembler makes extensive use of the cassette recorder.

A typical assembly programming session goes something like this:

Load the EDTASM tape. Write a program. Record the program on a second cassette tape. Record the corresponding machine language code on a third tape. Return to BASIC and run this system tape.

It doesn't work?

Reload the EDTASM tape. Reload the tape containing the program. Continue.

If you're like me, you couldn't wait to get hold of a disk system and throw the cassette recorder away. But without modification the assembler will not work with a disk. In fact, it can't even be loaded on a disk because it resides in the same area of memory used by DOS.

I am going to describe a new version of the editor/assembler. The original assembler can be used in bootstrap fashion to create this new version. Programs written on the modified assembler can be run without using the cassette or disk at all. Once written, programs can be saved on disk.

Additional commands convert the editor into a word processing system. This new editor/assembler requires at least 32K of memory.

Modifying EDTASM

Assume for a moment that the modification has been made. Enter the new assembler from the disk by typing EDTASM (ENTER). After a short pause, the words "TRS-80 EDITOR-ASSEMBLER 2.1" (or 2.2) will be printed on the screen. This new version has all the original commands. Refer to the instruction book for an explanation of them.

Several small modifications have been made. First of all, the command B returns control to DOS instead of BASIC. A keyboard debounce routine has been added. The text buffer is restricted to memory slots 5CF9-7FFF (or 5CF0-7FFF for version 1.2), which is the standard buffer for machines with 16K.

The function of the up arrow has been changed to display the previous page of text. The down arrow now prints the next line of text without an intervening star and makes it possible to go through the text line by line. The CLEAR key works.

A lowercase driver is available for readers who have added lowercase hardware. The printer routine has the ability to pause after each page. The BREAK key now works during printing. An optional serial printer driver is included.

There are also additional commands which will be convenient to discuss in groups.

Easier AL Programming

The assembler has two new commands designed to make assembly language programming easier:

```
M0 Method Zero.
M1 Method One.
```

The command M1 changes the method of text entry. Suppose we wish to enter the following:

```
LINE LD A,B ;COMMENT HERE
```

To do so, type LINE and press the space bar. The computer will tab to the next position. Type LD and press the space bar. Again the computer will tab to the following position. Type A,B and press spacebar. This time the computer will tab to the comment column and enter the semicolon comment prompt.

Now type the comment as usual. The space bar will no longer tab; instead it will resume its usual function. If an entire line is to be a comment, type ";" and the space bar will never act as a tab.

In this keyboard mode, backspacing past a tab can lead to unpredictable results. If a mistake in an earlier column is to be corrected, erase the entire line and start over or use EDIT.

The command M0 returns the editor to the original keyboard entry mode. When the system powers up, it will be in entry mode one.

No Tapes, No Disks

The assembler has four new commands which allow immediate execution of assembly language programs without using the cassette or the disk:

```
AM    Assemble into Memory.
PEEK  Examine Memory.
POKE  Modify Memory.
J     Jump.
```

Suppose the text buffer contains an assembly language program. The command AM works, essentially, like the usual ASSEMBLE. The computer prints an assembled version of the program on the screen and then types MOVE CODE? When the space bar is pressed, the machine language program will be entered into the computer instead of being loaded on tape.

This program must occupy memory in the range A000-BFFF hex. Otherwise the computer will print MEMORY TOO LOW or MEMORY TOO HIGH and return to the assembler prompt.(Readers with 48K may change this range to A000-FFFF.)

The J command is used to

transfer control to a machine language program created by AM. Thus, J A98C causes the computer to jump to location A98C (hex) and begin execution at that position. This command requires that the eddress be given in hex. The hex number should be four digits long and should not be followed by H. Thus, J 3C45, J BC11, and J 3542 are all correct es written.

A program that is to be executed by J should end with the instruction RET. This will return the essembler prompt * when the program is finished; the text buffer will remain intact.

The editor/assembler maintains a stack which can be used by the machine languege program. Of course, if the machine program initializes enother stack, the command RET will not return control to the assembler. In that case the program should end with the command JP 8F01H to return control to the editor/assembler. Initializing the stack is among the first things it does.

Programs executed with J may use BASIC ROM routines. A few users will want to use EDTASM routines instead. They should use JU (thus JU A2AD) instead of J and end programs which modify the stack pointer with JP 46DAH.

If it is necessery to exemine and modify memory, PEEK and POKE are used. Both of these commands are written for hex numbers. Thus PEEK 4D23 will cause the computer to print the contents of 4D23, and these contents will be printed in hex. Similarly, POKE 4D23,3E will enter the hex number 3E in memory location 4D23.

Disk Data Storage

The assembler has three commands designed to allow storage of data on the disk:

```
X            Move Text to High Memory.
Y            Move Text from High
             Memory.
AM + &&&&    Assemble into Memory Plus
             &&&&.
```

X moves the text buffer into memory locations 9CF0-BFFF. It also prints a message of the form (START = X'9CF0',END = X'A9CE') on the screen. Using this command, it is possible to load the text buffer onto disk as follows:

Enter X. Copy the displayed information about START and END on a piece of paper because the next step clears the screen. Enter B. You will find yourself in DOS mode. Enter:

```
DUMP FILENAME/CIM (START = X'9CF0',
  END = X'A9CE')
```

Naturally, the end statement varies from text to text.

At this point, you can return to the assembler with the text buffer intact. To do so, type EDTASM (ENTER) Y (ENTER).

Y moves the text buffer from locations 9CF0-BFFF down into the assembler. It is used to load text files from the disk into the essembler. To load such a file, return to DOS mode and enter LOAD FILENAME, EDTASM and Y.

Machine language programs are prepared for entry to the disk using the command AM. Suppose the text buffer contains an assembly language program which will be located somewhere between A000 and BFFF. Enter the corresponding machine code into the computer using AM. Return to DOS via the B command. Load the machine code onto the disk using a command similar to:

```
DUMP FILE/CMD (START = X'A02E',END =
  X'A02E',END = X'B13C',TRA = X'A111')
```

Occasionally you will want to write code which will not occupy memory In the interval A000-BFFF. The command AM + is used for this purpose. AM + 3C11, for instance, works just like AM except that the hex number 3C11 is added to each memory address before it is entered into the computer.

Suppose you want to write machine code starting at 7500. (All addresses here are in hex form.) This address is safely above the DOS addresses, but it is within the EDTASM area. Write the essembly code as ueual. Enter the comand AM + 2B10. (Notice that 2B10 Is A010 minus 7500.) Write the following program and assemble it into memory:

```
ORG 0A000H
LD   HL,0A010H
LD   DE,7500H
LD   BC,Number of Bytes in Program
LDIR
JP   0
```

Finally, enter the command J A000. The disk will turn on for e moment end you will find yourself in DOS mode with the machine language program in its intended position. Save it on disk as usual.

Word Processing

Finally, the assembler has two commands designed to convert the editor into a primitive word processing system. The word processing commands are:

```
M2     Method Two.
TYPE   Print Text.
```

The command M2 provides yet another method of entering text into the computer. With this method text can be entered continuously without the ENTER key. The computer will eutomatically issue line feeds at appropriate spots. Line feeds will occur between words when possible, but occasionally the computer will break a word in the middle.

M2 makes one other change. Shifted letters are not accepted under keyboard entry methods zero and one unless you add the lowercase modification. But shifted letters are accepted under method two. Such letters

Program Listing 1. EDTASM 1

```
                00100 ; INSTRUCTIONS:  Enter DOS.  Turn on DEBUG and use the M-command to
                00110 ; enter the code below.  Return to DOS, turn off DEBUG, and issue the
                00120 ; command BASIC2.  Answer MEMORY SIZE with RETURN.  Issue the SYSTEM
                00130 ; command.  Respond to the prompt with EDTASM and load the EDTASM
                00140 ; tape.  Respond to the second prompt with /40960.  You will
                00150 ; automatically return to DOS.  Enter
                00160 ; DUMP EDTASM1/CIM (START=X'7000',END=X'8A00').
                00170 ;
                00180 ;
                00190 ; TEMPORARY CODE TO PUT EDTASM ON DISK
                00200 ;
                00210 ;
A000            00220          ORG     0A000H
A000 F3         00230          DI
A001 210043     00240          LD      HL,4300H
A004 110070     00250          LD      DE,7000H
A007 01001A     00260          LD      BC,1A00H
A00A EDB0       00270          LDIR
A00C C30000     00280          JP      0000
0000            00290          END
00000 TOTAL ERRORS
```

will be displayed on the screen as ordinary capital letters, of course. We will see their significance shortly.

M2 affects all entry from the keyboard, including responses to the assembler prompt. This will make no difference unless you are in the habit of shifting letters at random. A shifted letter looks the same on the screen, but means something different to the computer. Convert back to the original keyboard entry method with M0 if you run into trouble.

The TYPE command works essentially like the editor/assembler command T; it outputs the text buffer to the printer.

The text in the buffer can be created by any of the keyboard entry methods, but from now on I will assume that text is entered with M2.

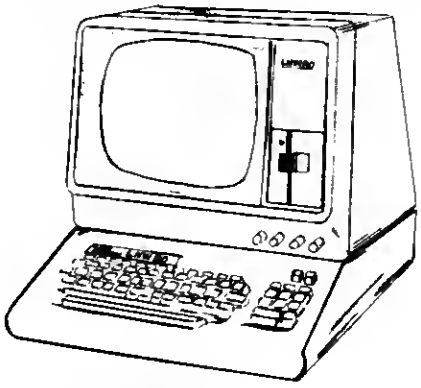
Text that will be printed first enters an internal buffer. When this buffer has enough text to fill a line, the computer searches backward from the end until it comes to a space. All characters before the space are printed and the remaining characters are placed at the beginning of the buffer for the next line.

continues to page 238

Program Listing 2. EDTASM 2

```
                00100 ; INSTRUCTIONS:  Enter DOS.  Load EDTASM1.  Turn on DEBUG and enter
                00110 ; the code below using the M-command.  Return to DOS, turn off DEBUG,
                00120 ; and enter DUMP EDTASM2/CMD (START=X'7000',END=X'8B30',TRA=X'8A00').
                00130 ;      There is an alternate way to do this.  Using the unmodified
                00140 ; EDTASM, assemble the code below and output it to cassette tape.
                00150 ; Enter DOS and load EDTASM1.  Enter BASIC and load the system tape
                00160 ; just created.  Return to DOS and dump EDTASM2 as before.
                00170 ;      You may test the result by running EDTASM2.  It has a smaller
                00180 ; than normal edit buffer and its cassette operations have been
                00190 ; temporarily disabled, but it will assemble code directly into
                00200 ; memory when you hit ENTER after the prompt READY CASSETTE.
                00210 ;
                00220 ;
                00230 ; CODE TO RELOCATE EDTASM IN CORRECT POSITION
                00240 ;
                00250 ;
8A00            00260          ORG     8A00H
8A00 F3         00270          DI                      ;Relocate EDTASM and run
8A01 210070     00280          LD      HL,7000H
8A04 110043     00290          LD      DE,4300R
8A07 01001A     00300          LD      BC,1A00H
8A0A EDB0       00310          LDIR
8A0C C38A46     00320          JP      468AH
                00330 ;
                00340 ;
                00350 ; CODE TO INTERCEPT TAPE OUTPUT AND ASSEMBLE DIRECTLY INTO MEMORY
                00360 ;
                00370 ;
8A0F E5         00380 ASSEM    PUSH    HL              ;Begin assembling code into memory
8A10 D5         00390          PUSH    DE
8A11 C5         00400          PUSH    BC
8A12 F5         00410          PUSH    AF
8A13 47         00420          LD      B,A             ;Current output byte
8A14 3A1B8B     00430          LD      A,(ADD+2)       ;Determine position in assembly cycle
8A17 FE00       00440          CP      0
8A19 2838       00450          JR      Z,ZERO
8A1B FE01       00460          CP      1
8A1D 283F       00470          JR      Z,ONE
8A1F FE02       00480          CP      2
8A21 2840       00490          JR      Z,TWO
8A23 FE03       00500          CP      3
8A25 283C       00510          JR      Z,TWO
8A27 FE04       00520          CP      4
8A29 2838       00530          JR      Z,TWO
8A2B FE05       00540          CP      5
8A2D 2834       00550          JR      Z,TWO
8A2F FE06       00560          CP      6
8A31 2830       00570          JR      Z,TWO
8A33 FE07       00580          CP      7
8A35 282C       00590          JR      Z,TWO
8A37 FE08       00600          CP      8
8A39 2831       00610          JR      Z,EIGHT
8A3B FE09       00620          CP      9
8A3D 2848       00630          JR      Z,NINE
8A3F FE0A       00640          CP      10
8A41 284A       00650          JR      Z,TEN
8A43 FE0B       00660          CP      11
8A45 2850       00670          JR      Z,ELE
8A47 FE0C       00680          CP      12
8A49 CAEE8A     00690          JP      Z,TWE
8A4C FE0D       00700          CP      13
8A4E CA098B     00710          JP      Z,THI
8A51 1814       00720          JR      END
8A53 78         00730 ZERO     LD      A,B             ;Leader and sync byte
8A54 FE00       00740          CP      0
8A56 280F       00750          JR      Z,END
```

THE FIRST TRS-80® COMPATIBLE COMPUTER WITH HIGH DENSITY COLOR GRAPHICS!



LNW RESEARCH introduces the LNW80, a high performance color computer, compatible with the TRS-80™ Model I. The fully integrated LNW80 is a sophisticated and versatile microcomputer with the following powerful features.

COMPATIBILITY

Hardware and software compatible to the Radio Shack TRS-80™ Model I computer, provides the widest software base of any microcomputer. Cassette interface; expansion bus.

DISPLAY

Quality upper and lower case display.

Two modes of color graphics, high resolution graphics, 384 x 192 in eight colors—higher density than the Apple II.* Low density color graphics of 128 x 192 are also available in eight colors.

High resolution—black and white graphics—of 384 x 192 mixed with text and TRS-80™ standard graphics.

Reverse video, composite video RF output.

PERFORMANCE

The LNW80 utilizes the fast Z-80A microprocessor which executes at a speed of 4 MHZ—over twice the speed of the TRS-80™ Model I.

LNW80

PC BOARD **$89.95**

Ask about our : Keyboard cabinet Leader VIDEO 100-80

LNW RESEARCH

LNW RESEARCH 3183-E AIRWAY AVE COSTA MESA CA 92626 714-552-8946

✓53

NEW — EXTERNAL DATA SEPARATOR — ASSEMBLED AND FULLY TESTED — $14.95 — SOME SOLDERING REQUIRED

SYSTEM EXPANSION

AT $69.95 [PC BOARD & USER MANUAL]

- SERIAL RS232C/20mA I/O
- FLOPPY CONTROLLER
- 32K BYTES MEMORY
- PARALLEL PRINTER PORT
- DUAL CASSETTE PORT
- REAL-TIME CLOCK
- SCREEN PRINTER BUS
- ONBOARD POWER SUPPLY
- SOFTWARE COMPATIBLE
- SOLDER MASK, SILK SCREEN

ORDERING INFORMATION

Add $3 for postage and handling. CA residents add 6% sales tax

VISA

*Apple II is a TM of Apple Computer Inc. TRS-80 is a TM of Tandy Corp.

TRS-80* — CONDENSE

The Ultimate in BASIC Compression Utilities

** Release 1.3 Now Available **

- Write BASIC programs using single statement lines for ease of maintenence.
- Write BASIC programe with unlimited remarks and comments to improve program readability and documentation....

— AND STILL GET —

OPTIMUM USE OF MEMORY — FASTER PROGRAM EXECUTION

- Compresssss programe up to 70% of original size
- Improves execution time by as much as 30%
- Creates multiple-statement program lines
- Blank compression
- Remark and comment deletion
- Renumbers GOTO, GOSUB, THEN, ELSE, and RESUME statements which reference deleted line numbers
- PLUS THESE NEW USER REQUESTED OPTIONS:
  - Retention of low numbered remark statements
  - Checkpoint / Restart Facilities
  - Phase 1 work file

Model I $21.95 (Diskette)

Model II $24.95 (Diskette)

INTERNATIONAL SOFTWARE ASSOCIATES

P.O. Box 14805 ✓187

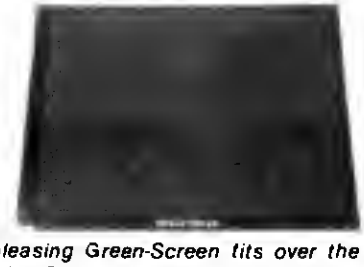
Omaha, Ne. 68124

Tandy Corporation*

---

for the TRS-80 from Micro-Mega

The Original GREEN-SCREEN



The eye-pleasing Green-Screen fits over the front of your TRS-80 Video Display and gives you improved contrast with reduced glare. You get bright luminous green characters and graphics like those featured by more expensive CRT units.

Don't confuse the Original Green-Screen with a piece of thin film stuck to the face of your video tube, such as that advertised by others. The Original Green-Screen is mounted in a full frame perfectly matched to the color and texture of the TRS-80 Video Display. It is attached with adhesive strips which do not mar your unit in any way.

The full frame design of the Original Green-Screen "squares off" the tece of your video display and greatly improves the overall appearance of your system.

THE GREEN-SCREEN.....*$13.95*

Add $1.00 for postage and handling.

Terms: Check or money order, no CODs or credit cards, please. Add amount shown for postage and handling to price of the item. All items shipped within 48 hours by first class or priority mail. Virginia residents, add 4% sales tax.

✓29

Micro-Mega · P.O. Box 6265 · Arlington, Va 22206

```
8A58 FEA5      00760          CP      0A5H
8A5A 2019      00770          JR      NZ,ERR
8A5C 1805      00780          JR      TWO
8A5E 78        00790 ONE      LD      A,B             ;Start byte
8A5F FE55      00800          CP      55H
8A61 2012      00810          JR      NZ,ERR
8A63 211B8B    00820 TWO      LD      HL,ADD+2        ;Name of tape
8A66 34        00830          INC     (HL)
8A67 F1        00840 END      POP     AF              ;Return to assembly routine
8A68 C1        00850          POP     BC
8A69 D1        00860          POP     DE
8A6A E1        00870          POP     HL
8A6B C9        00880          RET
8A6C 78        00890 EIGHT    LD      A,B             ;Beginning of code
8A6D FE3C      00900          CP      3CH
8A6F 28F2      00910          JR      Z,TWO
8A71 FE78      00920          CP      78H             ;or end of tape
8A73 280B      00930          JR      Z,DONE
8A75 217B8A    00940 ERR      LD      HL,ERRO         ;Print error message and end
8A78 C32B47    00950          JP      472BH
8A7B           00960 ERRO     EQU     $
8A7B 45        00970          DEFB    'E'
8A7C 52        00980          DEFB    'R'
8A7D 52        00990          DEFB    'R'
8A7E 4F        01000          DEFB    'O'
8A7F D2        01010          DEFB    0D2H
8A80 211B8B    01020 DONE     LD      HL,ADD+2        ;Ignore remaining output
8A83 360E      01030          LD      (HL),14
8A85 18E0      01040          JR      END
8A87 211F8B    01050 NINE     LD      HL,ADD+6        ;Number of code bytes to follow
8A8A 70        01060          LD      (HL),B
8A8B 18D6      01070          JR      TWO
8A8D 211C8B    01080 TEN      LD      HL,ADD+3        ;Low byte of code address
8A90 78        01090          LD      A,B
8A91 77        01100          LD      (HL),A
8A92 23        01110          INC     HL
8A93 23        01120          INC     HL
8A94 77        01130          LD      (HL),A
8A95 18CC      01140          JR      TWO
8A97 211D8B    01150 ELE      LD      HL,ADD+4        ;High byte of code address
8A9A 70        01160          LD      (HL),B
8A9B 23        01170          INC     HL
8A9C 7E        01180          LD      A,(HL)
8A9D 80        01190          ADD     A,B
8A9E 77        01200          LD      (HL),A
8A9F 21198B    01210          LD      HL,ADD
8AA2 5E        01220          LD      E,(HL)
8AA3 23        01230          INC     HL
8AA4 56        01240          LD      D,(HL)
8AA5 3A1D8B    01250          LD      A,(ADD+4)
8AA8 67        01260          LD      H,A
8AA9 3A1C8B    01270          LD      A,(ADD+3)
8AAC 6F        01280          LD      L,A
8AAD 19        01290          ADD     HL,DE
8AAE 221C8B    01300          LD      (ADD+3),HL
8AB1 18B0      01310          JR      TWO             ;Temporarily allow assembly anywhere
8AB3 7C        01320          LD      A,H
8AB4 FEA0      01330          CP      0A0H
8AB6 380D      01340          JR      C,ERR2
8AB8 FEBF      01350          CP      0BFH
8ABA 38A7      01360          JR      C,TWO
8ABC 200D      01370          JR      NZ,ERR3
8ABE 7D        01380          LD      A,L
8ABF FE80      01390          CP      80H
8AC1 3008      01400          JR      NC,ERR3
8AC3 189E      01410 OK1      JR      TWO
8AC5 21D18A    01420 ERR2     LD      HL,ERRO2
8AC8 C32B47    01430          JP      472BH
8ACB 21DF8A    01440 ERR3     LD      HL,ERRO3
8ACE C32B47    01450          JP      472BH
8AD1           01460 ERRO2    EQU     $
8AD1 4D        01470          DEFB    'M'
8AD2 45        01480          DEFB    'E'
8AD3 4D        01490          DEFB    'M'
8AD4 4F        01500          DEFB    'O'
8AD5 52        01510          DEFB    'R'
8AD6 59        01520          DEFB    'Y'
8AD7 20        01530          DEFB    ' '
8AD8 54        01540          DEFB    'T'
8AD9 4F        01550          DEFB    'O'
8ADA 4F        01560          DEFB    'O'
8ADB 20        01570          DEFB    ' '
8ADC 4C        01580          DEFB    'L'
```

Program continues

Reader Service—see page 274

```
8ADD 4F       01590          DEFB    'O'
8ADE D7       01600          DEFB    0D7H
8ADF          01610 ERRO3    EQU     $
8ADF 4D       01620          DEFB    'M'
8AE0 45       01630          DEFB    'E'
8AE1 4D       01640          DEFB    'M'
8AE2 4F       01650          DEFB    'O'
8AE3 52       01660          DEFB    'R'
8AE4 59       01670          DEFB    'Y'
8AE5 20       01680          DEFB    ' '
8AE6 54       01690          DEFB    'T'
8AE7 4F       01700          DEFB    'O'
8AE8 4F       01710          DEFB    'O'
8AE9 20       01720          DEFB    ' '
8AEA 48       01730          DEFB    'B'
8AEB 49       01740          DEFB    'I'
8AEC 47       01750          DEFB    'G'
8AED C8       01760          DEFB    0C8H
8AEE 78       01770 TWE      LD      A,B             ;Actual code
8AEF 2A1C8B   01780          LD      HL,(ADD+3)
8AF2 77       01790          LD      (HL),A
8AF3 23       01800          INC     HL
8AF4 221C8B   01810          LD      (ADD+3),HL
8AF7 211E8B   01820          LD      HL,ADD+5
8AFA 46       01830          LD      B,(HL)
8AFB 80       01840          ADD     A,B
8AFC 77       01850          LD      (HL),A
8AFD 23       01860          INC     HL
8AFE 7E       01870          LD      A,(HL)
8AFF 3D       01880          DEC     A
8B00 CA638A   01890          JP      Z,TWO
8B03 321F8B   01900          LD      (ADD+6),A
8B06 C3678A   01910          JP      END
8B09 78       01920 THI      LD      A,B             ;Check sum
8B0A 211E8B   01930          LD      HL,ADD+5
8B0D BE       01940          CP      (HL)
8B0E C2758A   01950          JP      NZ,ERR
8B11 3E08     01960          LD      A,8
8B13 321B8B   01970          LD      (ADD+2),A
8B16 C3678A   01980          JP      END
8B19          01990 ADD      EQU     $
8B19 00       02000          DEFB    0               ;Low byte of addition to memory
8B1A 00       02010          DEFB    0               ;High byte of addition to memory
8B1B 00       02020          DEFB    0               ;Position in assembly cycle
8B1C 00       02030          DEFB    0               ;Low byte of current memory
8B1D 00       02040          DEFB    0               ;High byte of current memory
8B1E 00       02050          DEFB    0               ;Check sum
8B1F 00       02060          DEFB    0               ;Number of bytes
8B20          02070 CONT     EQU     $
              02080 ;
              02090 ;
              02100 ; CODE TO SET TOP OF EDIT BUFFER
              02110 ;
              02120 ;
7395          02130          ORG     7395H
7395 21FF6F   02140          LD      HL,6FFFH        ;Temporary value; eventually 7FFFH
7398 221341   02150          LD      (4113H),HL
739B C3A246   02160          JP      46A2H
              02170 ;
              02180 ;
              02190 ; TEMPORARY CODE SENDING CASSETTE OUTPUT TO ABOVE ASSEMBLY ROUTINE
              02200 ;
              02210 ;
703D          02220          ORG     703DH
703D C9       02230          DEFB    0C9H
7089          02240          ORG     7089H
7089 C30F8A   02250          JP      ASSEM
              02260 ;
              02270 ;
              02280 ; TEMPORARY CODE INITIALIZING ASSEMBLER
              02290 ;
              02300 ;
73DA          02310          ORG     73DAH           ;Start of cleanup before EDTASM prompt
73DA C3208B   02320          JP      CONT
8B20          02330          ORG     CONT
8B20 31FE42   02340          LD      SP,42FEH        ;Command replaced by JP
8B23 211B8B   02350          LD      HL,ADD+2        ;Assembler cycle position
8B26 3600     02360          LD      (HL),0
8B28 C3DD46   02370          JP      46DDH
              02380 ;
              02390 ;
0000          02400          END
00000 TOTAL ERRORS
```

Program Listing 3. EDTASM 3

```
                00100 ; INSTRUCTIONS:  Run EDTASM2.  Enter the text below and assemble it.
                00110 ; Answer the prompt READY CASSETTE with ENTER; the revisions will
                00120 ; be entered directly into the computer.  The new code affects the
                00130 ; running version of EDTASM adversely, so immediately
                00140 ; return to DOS via the B command and enter
                00150 ; DUMP EDTASM3/CMD (START=X'7000',END=X'8C5E',TRA=X'8A00').
                00160 ;      You may test the new version of EDTASM.  The cassette
                00170 ; operations are reenabled.  From now on, you must use the command
                00180 ; AM to assemble directly into memory.
                00190 ;
                00200 ;
                00210 ; CODE INITIALIZING ASSEMBLER BEFORE EDTASM PROMPT
                00220 ;
                00230 ;
                00240 ; Owners of Version 1.2 should change ASSEM1 to 51E7H.
51E3            00250 ASSEM1  EQU     51E3H
8820            00260 CONT    EQU     8B20H
8B19            00270 ADD     EQU     8B19H
8A0F            00280 ASSEM   EQU     8A0FH
8B20            00290         ORG     CONT            ;Start of cleanup before prompt
8B20 31FE42     00300         LD      SP,42FEH        ;Command replaced by JP
8B23 21198B     00310         LD      HL,ADD          ;Initialize assembler
8B26 3600       00320         LD      (HL),0
8B28 23         00330         INC     HL
8B29 3600       00340         LD      (HL),0
8B2B 23         00350         INC     HL
8B2C 3600       00360         LD      (HL),0
8B2E 213D43     00370         LD      HL,433DH        ;Enable cassette operations
8B31 36E5       00380         LD      (HL),0E5H
8B33 218943     00390         LD      HL,4389H
8B36 36E5       00400         LD      (HL),0E5H
8B38 23         00410         INC     HL
8B39 36C5       00420         LD      (HL),0C5H
8B3B 23         00430         INC     HL
8B3C 36D5       00440         LD      (HL),0D5H
8B3E AF         00450         XOR     A               ;Method zero for EDTASM
8B3F 32458B     00460         LD      (TYPE),A
8B42 C3D046     00470         JP      46DDH
8B45 00         00480 TYPE    DEFB    0               ;Method currently used
8B46 01         00490         DEFB    1               ;Method chosen by M command
                00500 ;
                00510 ;
                00520 ; ADDITIONAL COMMANDS
                00530 ;
                00540 ;
8B47            00550 COMM    EQU     $
7428            00560         ORG     7428H
7428 C3478B     00570         JP      COMM
8B47            00580         ORG     COMM
8B47 215E8B     00590         LD      HL,TABLE
8B4A 0606       00600         LD      B,6
8B4C BE         00610 NEXT1   CP      (HL)            ;First letter of command
8B4D 23         00620         INC     HL
8B4E 5E         00630         LD      E,(HL)          ;Low byte
8B4F 23         00640         INC     HL
8B50 56         00650         LD      D,(HL)          ;and high byte of address
8B51 23         00660         INC     HL
8B52 2002       00670         JR      NZ,MORE
8B54 D5         00680         PUSH    DE
8B55 C9         00690         RET
8B56 10F4       00700 MORE    DJNZ    NEXT1
8B58 21A247     00710         LD      HL,47A2H        ;Error
8B5B C32B47     00720         JP      472BH
8B5E 4D         00730 TABLE   DEFB    'M'             ;Method command
8B5F 708B       00740         DEFW    METHOD
8B61 00         00750 X       DEFB    0               ;Eventually X
8B62 00         00760         DEFB    0
8B63 00         00770         DEFB    0
8B64 00         00780 Y       DEFB    0               ;Eventually Y
8B65 00         00790         DEFB    0
8B66 00         00800         DEFB    0
8B67 00         00810 J       DEFB    0               ;Eventually J
8B68 00         00820         DEFB    0
8B69 00         00830         DEFB    0
8B6A 00         00840         DEFB    0               ;Available for future expansion
8B6B 00         00850         DEFB    0
8B6C 00         00860         DEFB    0
8B6D 00         00870         DEFB    0               ;Available for future expansion
8B6E 00         00880         DEFB    0
8B6F 00         00890         DEFB    0
                00900 ;
                00910 ;
```

Program continues

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION (Required by 39 U.S.C. 3685). 1. Title of publication, *80 Microcomputing*. 2. Date of filing, Oct. 1, 1980. 3. Frequency of issue, Monthly. A. No. of issues published annually, 12. B. Annual subscription price, $18.00. 4. Location of known office of publication (Street, City, County, State and ZIP Code) (Not printers), 80 Pine Street, Peterborough, Hillsboro County, N.H. 03458. 5. Location of the headquarters or general business offices of the publishers (Not printers), 80 Pine Street, Peterborough, Hillsboro County, N.H. 03458. 6. Names and complete addresses of publisher, editor and managing editor. Publisher (Name and Address), Wayne Green, Peterborough, N.H. 03458. Editor (Name and Address), Wayne Green, Peterborough, N.H. 03458. Managing Editor (Name and Address), Michael Comendul, PO Box 343, Antrim, N.H. 03440. 7. Owner (If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given. If the publication is published by a nonprofit organization, its name and address must be stated.) Name, 1001001, Inc., Peterborough, N.H. 03458. Wayne Green, Peterborough, N.H. 03458. 8. Known bondholders, mortgagees and other security holders owning or holding 1 percent or more of total amount of bonds, mortgages or other securities (If there are none, so state) Name, none. 9. For completion by nonprofit organizations authorized to mail at special rates (Section 132.122, PSM) The purpose, function and nonprofit status of this organization and the exempt status for Federal income tax purposes (Check one) Not applicable. 10. Extent and nature of circulation. (X) Average No. copies each issue during preceding 12 months. (Y) Actual No. copies of single issue published nearest to filing date. A. Total No. of copies printed (Net Press Run) (X) 54,660 (Y) 63,492. B. Paid circulation 1. Sales through dealers and carriers, street vendors and counter sales, (X) 9,897 (Y) 11,315. 2. Mail subscriptions (X) 32,352 (Y) 43,168. C. Total paid circulation (Sum of 10B1 and 10B2) (X) 42,249 (Y) 54,483. D. Free distribution by mail, carrier or other means samples, complimentary, and other free copies (X) 244 (Y) 398. E. Total distribution (Sum of C and D) (X) 42,493 (Y) 54,881. F. Copies not distributed 1. Office use, left over, unaccounted, spoiled after printing (X) 11,270 (Y) 7,714. 2. Returns from news agents (X) 897 (Y) 897. G. Total (Sum of E, F1 and 2—should equal net press run shown in A) (X) 54,660 (Y) 63,492. 11. I certify that the statements made by me above are correct and complete. Signature and title of editor, publisher, business manager, or owner. Debra Boudreau, Business Manager.

```
                00920 ; CHOOSE KEYBOARD ENTRY METHOD
                00930 ;
                00940 ;
8B70 CDBB49     00950 METHOD  CALL    49BBH           ;Get next buffer letter
8B73 2812       00960         JR      Z,OH
8B75 0600       00970         LD      B,0
8B77 FE30       00980         CP      30H
8B79 280F       00990         JR      Z,FIX1
8B7B 0601       01000         LD      B,1
8B7D FE31       01010         CP      31H
8B7F 2809       01020         JR      Z,FIX1
8B81 0602       01030         LD      B,2
8B83 FE32       01040         CP      32H
8B85 2807       01050         JR      Z,FIX2
8B87 C3F4BB     01060 OH      JP      BAD             ;Error
8B8A 3E61       01070 FIX1    LD      A,97            ;Upper case only
8B8C 1802       01080         JR      OHH
8B8E 3E80       01090 FIX2    LD      A,128           ;Lower case
8B90 324A46     01100 OHH     LD      (464AH),A
8B93 78         01110         LD      A,B
8B94 32468B     01120         LD      (TYPE+1),A
8B97 C9         01130         RET
                01140 ;
                01150 ;
                01160 ; CODE TO DETECT AM COMMAND
                01170 ;
                01180 ;
8B9B            01190 AM      EQU     $
7618            01200         ORG     7618H
7618 988B       01210         DEFW    AM
8B98            01220         ORG     AM
8B98 E5         01230         PUSH    HL
8B99 D5         01240         PUSH    DE
8B9A C5         01250         PUSH    BC
8B9B F5         01260         PUSH    AF
8B9C CDBB49     01270         CALL    49BBH           ;Next command letter
8B9F 2807       01280         JR      Z,ABORT         ;No next letter
8BA1 FE4D       01290         CP      4DH             ;M
```

Program continues

```
8BA3 2816      01300          JR      Z,NEXT
8BA5 CDAF8B    01310 ABORT1   CALL    RETN
8BA8 F1        01320 ABORT    POP     AF
8BA9 C1        01330          PDP     BC
8BAA D1        01340          POP     DE
8BAB E1        01350          PDP     HL
8BAC C3E351    01360          JP      ASSEM1          ;Assembler routine
8BAF 21AA41    01370 RETN     LD      HL,41AAH        ;Return letter to input buffer
8BB2 34        01380          INC     (HL)
8BB3 2AA841    01390          LD      HL,(41A8H)
8BB6 2B        01400          DEC     HL
8BB7 22A841    01410          LD      (41A8H),HL
8BBA C9        01420          RET
8BBB 213D43    01430 NEXT     LD      HL,433DH        ;Cassette output to ASSEM routine
8BBE 36C9      01440          LD      (HL),0C9H
8BC0 218943    01450          LD      HL,4389H
8BC3 36C3      01460          LD      (HL),0C3H
8BC5 210FBA    01470          LD      HL,ASSEM
8BC8 228A43    01480          LD      (438AH),HL
8BCB CDBB49    01490          CALL    49BBH           ;Next letter
8BCE 28D8      01500          JR      Z,ABORT         ;No letter
8BD0 FE2B      01510          CP      2BH             ;+
8BD2 20D1      01520          JR      NZ,ABORT1       ;Replace letter
8BD4 CDE28B    01530          CALL    ROUT            ;High byte of addition to memory
8BD7 321A8B    01540          LD      (ADD+1),A
8BDA CDE2BB    01550          CALL    ROUT            ;Low byte of addition to memory
8BDD 3219BB    01560          LD      (ADD),A
8BE0 1BC6      01570          JR      ABORT           ;Go to assembler
8BE2 CDF08B    01580 ROUT     CALL    GET             ;Next letter
8BE5 57        01590          LD      D,A
8BE6 CDF0BB    01600          CALL    GET             ;Next letter
8BE9 5F        01610          LD      E,A
8BEA CDFA8B    01620          CALL    CONV            ;Convert ascii DE to number
8BED 2805      01630          JR      Z,BAD           ;DE not correct
8BEF C9        01640          RET
8BF0 CDBB49    01650 GET      CALL    49BBH           ;Next letter
8BF3 C0        01660          RET     NZ
8BF4 21A247    01670 BAD      LD      HL,47A2H        ;Error
```

Program continues

```
8BF7 C32B47   01680          JP      472BH
8BFA C5       01690 CONV     PUSH    BC          ;Convert ascii in DE to number in
8BFB 7A       01700          LD      A,D         ;A and set zero flag if error
8BFC CD138C   01710          CALL    CONV1
8BFF 280F     01720          JR      Z,BAD1
8C01 57       01730          LD      D,A
8C02 7B       01740          LD      A,E
8C03 CD138C   01750          CALL    CONV1
8C06 2808     01760          JR      Z,BAD1
8C08 0610     01770          LD      B,16
8C0A 82       01780 ADA      ADD     A,D
8C0B 10FD     01790          DJNZ    ADA
8C0D 04       01800          INC     B           ;Reset zero flag
8C0E C1       01810          POP     BC
8C0F C9       01820          RET
8C10 AF       01830 BAD1     XOR     A           ;Set zero flag
8C11 C1       01840          POP     BC
8C12 C9       01850          RET
8C13 E5       01860 CONV1    PUSH    HL          ;Convert ascii in A to number
8C14 C5       01870          PUSH    BC          ;in A and set zero flag if error
8C15 212B8C   01880          LD      HL,TABLE1
8C18 0610     01890          LD      B,16
8C1A BE       01900 LOOP     CP      (HL)
8C1B 2808     01910          JR      Z,YES
8C1D 23       01920          INC     HL
8C1E 23       01930          INC     HL
8C1F 10F9     01940          DJNZ    LOOP
8C21 AF       01950          XOR     A           ;Set zero flag
8C22 C1       01960          POP     BC
8C23 E1       01970          POP     HL
8C24 C9       01980          RET
8C25 23       01990 YES      INC     HL
8C26 7E       02000          LD      A,(HL)
8C27 04       02010          INC     B           ;Reset zero flag
8C28 C1       02020          POP     BC
8C29 E1       02030          POP     HL
8C2A C9       02040          RET
```

Program continues

```
8C2B 30          02050 TABLE1   DEFB    '0'
8C2C 00          02060          DEFB    0
8C2D 31          02070          DEF8    '1'
8C2E 01          02080          DEFB    1
8C2F 32          02090          DEFB    '2'
8C30 02          02100          DEFB    2
8C31 33          02110          DEFB    '3'
8C32 03          02120          DEFB    3
8C33 34          02130          DEFB    '4'
8C34 04          02140          DEFB    4
8C35 35          02150          DEFB    '5'
8C36 05          02160          DEFB    5
8C37 36          02170          DEFB    '6'
8C38 06          02180          DEFB    6
8C39 37          02190          DEFB    '7'
8C3A 07          02200          DEFB    7
8C3B 38          02210          DEFB    '8'
8C3C 08          02220          DEFB    8
8C3D 39          02230          DEFB    '9'
8C3E 09          02240          DEFB    9
8C3F 41          02250          DEFB    'A'
8C40 0A          02260          DEFB    10
8C41 42          02270          DEFB    'B'
8C42 0B          02280          DEFB    11
8C43 43          02290          DEFB    'C'
8C44 0C          02300          DEFB    12
8C45 44          02310          DEFB    'D'
8C46 0D          02320          DEFB    13
8C47 45          02330          DEFB    'E'
8C48 0E          02340          DEFB    14
8C49 46          02350          DEFB    'F'
8C4A 0F          02360          DEFB    15
                 02370 ;
                 02380 ;
                 02390 ; DEBOUNCE
                 02400 ;
```


✓ Reader Service—see page 274

```
                02410 ;
8C4B            02420 DEB       EQU     $
7107            02430           ORG     7107H
7107 C34B8C     02440           JP      DEB
8C4B            02450           ORG     DEB
8C4B C5         02460           PUSH    BC
8C4C 01DF04     02470           LD      BC,4DFH
8C4F 0B         02480 HERE      DEC     BC
8C50 78         02490           LD      A,B
8C51 B1         02500           OR      C
8C52 20FB       02510           JR      NZ,HERE
8C54 C1         02520           POP     BC
8C55 0A         02530           LD      A,(BC)
8C56 A3         02540           AND     E
8C57 C8         02550           RET     Z
8C58 7A         02560           LD      A,D
8C59 07         02570           RLCA
8C5A 07         02580           RLCA
8C5B C30A44     02590           JP      440AH
                02600 ;
                02610 ;
                02620 ; CHANGE 'READY CASSETTE' TO 'MOVE CODE'
                02630 ;
                02640 ;
8C5E            02650 COHT1     EQU     $
75E4            02660           ORG     75E4H
75E4 4D         02670           DEFB    'M'
75E5 4F         02680           DEFB    'O'
75E6 56         02690           DEFB    'V'
75E7 45         02700           DEFB    'E'
75E8 20         02710           DEFB    ' '
75E9 43         02720           DEFB    'C'
75EA 4F         02730           DEFB    'O'
75EB 44         02740           DEFB    'D'
75EC 45         02750           DEFB    'E'
75ED BF         02760           DEFB    '?'+128
0000            02770           END
00000 TOTAL ERRORS
```


Program Listing 4. EDTASM 4

```
                00100 ; INSTRUCTIONS:  Run EDTASM3, enter the text below, and assemble
                00110 ; it into memory using AM.  Return to DOS and enter
                00120 ; DUMP EDTASM4/CMD (START=X'7000',END=X'8D66',TRA=X'8A00').
                00130 ;
                00140 ;
                00150 ; CLEAR KEY
                00160 ;
                00170 ;
                00180 ; Owners of Version 1.2 should change UP1 to 4C76H, DOWN1 to
                00190 ; 4C7BH, and ED to 4DC5H.
4C72            00200 UP1       EQU     4C72H
4C74            00210 DOWN1     EQU     4C74H
4DC1            00220 ED        EQU     4DC1H
8C5E            00230 CDHT1     EQU     8C5EH
8B45            00240 TYPE      EQU     8B45H
7663            00250           ORG     7663H
7663 C3         00260           DEFB    0C3H
7664 5E8C       00270           DEFW    CONT1
8C5E            00280           ORG     CONT1
8C5E FE5B       00290           CP      5BH             ;Instruction replaced by JP
8C60 CA6B49     00300           JP      Z,496BH
8C63 FE1F       00310           CP      1FH             ;Clear key
8C65 C26749     00320           JP      NZ,4967H
8C68 CD6E8C     00330           CALL    CLEAR
8C6B C3DA46     00340           JP      46DAH
8C6E 3E1C       00350 CLEAR     LD      A,1CH           ;Routine to clear screen
8C70 CD3947     00360           CALL    4739H
8C73 3E1F       00370           LD      A,1FH
8C75 CD3947     00380           CALL    4739H
8C78 3E0E       00390           LD      A,0EH
8C7A CD3947     00400           CALL    4739H
8C7D C9         00410           RET
                00420 ;
                00430 ;
                00440 ; DOWN ARROW
                00450 ;
                00460 ;
```

Program continues

```
8C7E            00470 DOWN    EQU     $
7624            00480         ORG     7624H
7624 7E8C       00490         DEFW    DOWN
8C7E            00500         ORG     DOWN
8C7E 3E0F       00510         LD      A,0FH           ;Cursor off
8C80 CD3947     00520         CALL    4739H
8C83 3E18       00530         LD      A,18H           ;Upward linefeed
8C85 CD3947     00540         CALL    4739H
8C88 CD744C     00550         CALL    DOWN1           ;Unmodified commsnd
8C8B 3E0E       00560         LD      A,0EH           ;Cursor on
8C8D C33947     00570         JP      4739H
                00580 ;
                00590 ;
                00600 ; UP ARROW
                00610 ;
                00620 ;
8C90            00630 UP      EQU     $
7621            00640         ORG     7621H
7621 908C       00650         DEFW    UP
8C90            00660         ORG     UP
8C90 061C       00670         LD      B,28
8C92 C5         00680 LOOP1   PUSH    BC
8C93 CD724C     00690         CALL    UP1             ;Previous line
8C96 C1         00700         POP     BC
8C97 10F9       00710         DJNZ    LOOP1
8C99 CD6E8C     00720         CALL    CLEAR
8C9C C32A4C     00730         JP      4C2AH           ;Print page
                00740 ;
                00750 ;
                00760 ; SET E=0 AT START OF LINE ENTER ROUTINE
                00770 ;
                00780 ;
8C9F            00790 E1      EQU     $
7644            00800         ORG     7644H
7644 C3         00810         DEFB    0C3H
7645 9F8C       00820         DEFW    E1
8C9F            00830         ORG     E1
8C9F 22A841     00840         LD      (41A8H),HL      ;Instruction replaced by JP
8CA2 1E00       00850         LD      E,0
8CA4 C34749     00860         JP      4947H
                00870 ;
                00880 ;
                00890 ; CONVERT TO CORRECT METHOD AT START OF INSTRUCTION
                00900 ;
                00910 ;
8CA7            00920 C1      EQU     $
73FB            00930         ORG     73FBH
73FB C3         00940         DEFB    0C3H
73FC A78C       00950         DEFW    C1
8CA7            00960         ORG     C1
8CA7 3A468B     00970         LD      A,(TYPE+1)      ;Nethod to be used
8CAA 324588     00980         LD      (TYPE),A        ;Active method
8CAD CD8849     00990         CALL    4988H           ;Instruction replaced by JP
8CB0 C3FE46     01000         JP      46FEH
                01010 ;
                01020 ;
                01030 ; REVISE KEYBOARD ENTRY NETHOD
                01040 ;
                01050 ;
8CB3            01060 K       EQU     $
7681            01070         ORG     7681H
7681 C3         01080         DEFB    0C3H
7682 B38C       01090         DEFW    K
8CB3            01100         ORG     K
8CB3 3A4588     01110         LD      A,(TYPE)
8CB6 FE00       01120         CP      0
8CB8 2860       01130         JR      Z,BACK
8CBA FE01       01140         CP      1
8CBC 2063       01150         JR      NZ,TWO2
8CBE 7E         01160         LD      A,(HL)          ;Letter from keyboard
8CBF FE3B       01170         CP      3BH             ;";"
8CC1 2004       01180         JR      NZ,EXT
8CC3 1E04       01190         LD      E,4             ;Stop fiddling around
8CC5 1853       01200         JR      BACK
8CC7 FE27       01210 EXT     CP      27H             ;"'"
8CC9 2004       01220         JR      NZ,EXT1
8CCB 1E04       01230         LD      E,4
8CCD 184B       01240         JR      BACK
8CCF FE20       01250 EXT1    CP      ' '
8CD1 2815       01260         JR      Z,EXT3
8CD3 FE09       01270         CP      9               ;Tab
8CD5 2008       01280         JR      NZ,EXT2
8CD7 1C         01290         INC     E               ;Next column
```

Program continues

```
8CD8 3E00      01300 NEW      LD      A,0
8CDA 32208D    01310          LD      (NUMBER),A      ;Count letters in column
8CDD 1838      01320          JR      BACK
8CDF 3A208D    01330 EXT2     LD      A,(NUMBER)      ;Additional letter in column
8CE2 3C        01340          INC     A
8CE3 32208D    01350          LD      (NUMBER),A
8CE6 1832      01360          JR      BACK
8CE8 1C        01370 EXT3     INC     E               ;New column
8CE9 7B        01380          LD      A,E
8CEA FE03      01390          CP      3
8CEC 3004      01400          JR      NC,EXT4         ;Already in third or fourth column
8CEE 3609      01410          LD      (HL),9          ;Tab instead
8CF0 18E6      01420          JR      NEW
8CF2 FE03      01430 EXT4     CP      3
8CF4 2802      01440          JR      Z,EXT5          ;In third column now
8CF6 1822      01450          JR      BACK
8CF8 1C        01460 EXT5     INC     E
8CF9 3609      01470          LD      (HL),9          ;Tab
8CFB 3E09      01480          LD      A,9
8CFD CD3947    01490          CALL    4739H           ;Print on screen
8D00 3A208D    01500          LD      A,(NUMBER)
8D03 FE08      01510          CP      8
8D05 3010      01520          JR      NC,EXT6         ;At least 8 characters in column
8D07 23        01530          INC     HL
8D08 3609      01540          LD      (HL),9          ;Another tab
8D0A 3E09      01550          LD      A,9
8D0C CD3947    01560          CALL    4739H           ;Print on screen
8D0F 23        01570          INC     HL
8D10 14        01580          INC     D
8D11 14        01590 KK       INC     D
8D12 3E3B      01600          LD      A,3BH           ;';'
8D14 C36249    01610          JP      4962H
8D17 23        01620 EXT6     INC     HL
8D18 18F7      01630          JR      KK
8D1A 7E        01640 BACK     LD      A,(HL)
8D1B 23        01650          INC     HL
8D1C 14        01660          INC     D
8D1D C38449    01670          JP      4984H
8D20 00        01680 NUMBER   DEFB    0               ;Number of chsracters in column
8D21 1C        01690 TWO2     INC     E
8D22 7B        01700          LD      A,E
8D23 FE39      01710          CP      39H             ;If end of line, make line feed
8D25 3806      01720          JR      C,EXT7
8D27 7E        01730          LD      A,(HL)          ;Current character
8D28 CD3947    01740          CALL    4739H           ;Print on screen
8D2B 1814      01750          JR      LINE            ;Line feed
8D2D 7E        01760 EXT7     LD      A,(HL)
8D2E FE20      01770          CP      ' '
8D30 280A      01780          JR      Z,EXT8
8D32 FE09      01790          CP      9               ;Tab
8D34 20E4      01800          JR      NZ,BACK
8D36 7B        01810          LD      A,E
8D37 C607      01820          ADD     A,7
8D39 5F        01830          LD      E,A
8D3A 18DE      01840          JR      BACK
8D3C 78        01850 EXT8     LD      A,E
8D3D FE30      01860          CP      30H             ;Look for line feed after 48 characters
8D3F 38D9      01870          JR      C,BACK
8D41 23        01880 LINE     INC     HL
8D42 14        01890          INC     D
8D43 C37549    01900          JP      4975H           ;Line feed
               01910 ;
               01920 ;
               01930 ; REVISE KEYBOARD ENTRY TO ACCOUNT FOR BACKSPACE
               01940 ;
               01950 ;
8D46           01960 R1       EQU     $
769D           01970          ORG     769DH
769D C3        01980          DEFB    0C3H
769E 468D      01990          DEFW    R1
8D46           02000          ORG     R1
8D46 3A458B    02010          LD      A,(TYPE)
8D49 FE02      02020          CP      2
8D4B 2001      02030          JR      NZ,S1
8D4D 1D        02040          DEC     E               ;Backspace
8D4E 15        02050 S1       DEC     D               ;Instructions replaced by JP
8D4F E1        02060          POP     HL
8D50 C9        02070          RET
8D51           02080 R2       EQU     $
76B8           02090          ORG     76B8H
76B8 C3        02100          DEFB    0C3H
76B9 518D      02110          DEFW    R2
8D51           02120          ORG     R2
```

Program continues

```
8D51 3A4588    02130          LD      A,(TYPE)
8D54 FE02      02140          CP      2
8D56 2004      02150          JR      NZ,S2
8D58 7B        02160          LD      A,E             ;Backspace past tab
8D59 D607      02170          SUB     7
8D5B 5F        02180          LD      E,A
8D5C C1        02190 S2       POP     BC              ;Instructions replaced by JP
8D5D E1        02200          POP     HL
8D5E C9        02210          RET
               02220 ;
               02230 ;
               02240 ; USE METROD ZERO FOR EDIT COMMAND
               02250 ;
               02260 ;
8D5F           02270 EDITOR   EQU     $
7627           02280          ORG     7627M
7627 5F8D      02290          DEFW    EDITOR
8D5F           02300          ORG     EDITOR
8D5F AF        02310          XOR     A
8D60 3245BB    02320          LD      (TYPE),A
8D63 C3C14D    02330          JP      ED
8D66           02340 CONT2    EQU     $
0000           02350          END
00000 TOTAL ERRORS
```

Program Listing 5. EDTASM 5

```
               00100 ; INSTRUCTIONS:  Run EDTASM4, enter the text below, and assemble
               00110 ; it to memory.  Return to DOS and enter BASIC2; answer MEMORY SIZE
               00120 ; with RETURN.  Issue the SYSTEM command and respond to the prompt
               00130 ; with /40960.  You will automatically return to DOS.  Enter
               00140 ; DUMP EDTASM5/CMD (START=X'7000',END=X'9236',TRA=X'8A00').
               00150 ;
               00160 ;
               00170 ; ENABLE P COMMAND WHEN BUFFER EMPTY
               00180 ;
               00190 ;
               00200 ; Owners of Version 1.2 should change BEGIN to 5CF0H and SIZE to
               00210 ; 1310H.
5CF9           00220 BEGIN    EQU     5CF9H
1307           00230 SIZE     EQU     1307H
8D5F           00240 EDITOR   EQU     8D5FH
8D66           00250 CONT2    EQU     8D66H
8C2B           00260 TABLE1   EQU     8C2BH
8BAF           00270 RETN     EQU     8BAFH
8B61           00280 X        EQU     8B61H
8B64           00290 Y        EQU     8B64H
8B67           00300 J        EQU     8B67H
7605           00310          ORG     7605H
7605 45        00320          DEFB    'E'             ;Interchange E
7606 5F8D      00330          DEFW    EDITOR
7626           00340          ORG     7626H
7626 50        00350          DEFB    'P'             ;and P commands
7627 668D      00360          DEFW    CONT2
741F           00370          ORG     741FH
741F 05        00380          DEFB    5               ;Additional command with buffer empty
               00390 ;
               00400 ;
               00410 ; PEEK AND POKE
               00420 ;
               00430 ;
8D66           00440          ORG     CONT2
8D66 CDBB49    00450          CALL    49BBH           ;Next letter
8D69 280B      00460          JR      Z,ABORT2
8D6B FE4F      00470          CP      'O'
8D6D 2825      00480          JR      Z,POKE
8D6F FE45      00490          CP      'E'
8D71 2841      00500          JR      Z,PEEK
8D73 CDAF8B    00510          CALL    RETN            ;Return letter to buffer
8D76 2A1541    00520 ABORT2   LD      HL,(4115H)      ;Buffer empty?
8D79 11F95C    00530          LD      DE,BEGIN
8D7C CDC24B    00540          CALL    4BC2H           ;DE = BC?
8D7F 2006      00550          JR      NZ,PRI
8D81 21C647    00560          LD      HL,47C6H        ;No text in buffer
8D84 C32B47    00570          JP      472BH
8D87 C32A4C    00580 PRI      JP      4C2AH           ;Print routine
8D8A CDBB49    00590 LETTER   CALL    49BBH           ;Next letter
8D8D C0        00600          RET     NZ
```

```
BD8E 21A247   00610 ERR4   LD     HL,47A2H      ;Error
8D91 C32B47   00620        JP     472BH
BD94 CD8A8D   00630 POKE   CALL   LETTER
8D97 FE4B     00640        CP     'K'
8D99 20F3     00650        JR     NZ,ERR4
BD9B CD8ABD   00660        CALL   LETTER
8D9E FE45     00670        CP     'E'
8DA0 20EC     00680        JR     NZ,ERR4
8DA2 CD098E   00690        CALL   SUB4          ;HL = address
8DA5 CD8A8D   00700        CALL   LETTER
8DA8 FE2C     00710        CP     ','
8DAA 20E2     00720        JR     NZ,ERR4
8DAC CDD48D   00730        CALL   SUB1          ;DE = ascii of value
8DAF CDE18D   00740        CALL   SUB2          ;A = value
BDB2 77       00750        LD     (HL),A
8DB3 C9       00760        RET
BDB4 CD8A8D   00770 PEEK   CALL   LETTER
BDB7 FE45     00780        CP     'E'
8DB9 20D3     00790        JR     NZ,ERR4
8DBB CD8A8D   00800        CALL   LETTER
BDBE FE4B     00810        CP     'K'
8DC0 20CC     00820        JR     NZ,ERR4
8DC2 CD098E   00830        CALL   SUB4          ;HL = address
BDC5 7E       00840        LD     A,(HL)        ;Value obtained by PEEK
BDC6 CD1BBE   00850        CALL   CONV2         ;HL = ascii of A
BDC9 7C       00860        LD     A,H
8DCA CD3947   00870        CALL   4739H         ;Print on screen
BDCD 7D       00880        LD     A,L
8DCE CD3947   00890        CALL   4739H         ;Print on screen
BDD1 CDD746   00900        CALL   46D7H         ;Line feed and return to EDTASM
BDD4 CDBB49   00910 SUB1   CALL   49BBH         ;Subroutine gets two letters from
8DD7 28B5     00920        JR     Z,ERR4        ;input buffer and loads them
8DD9 57       00930        LD     D,A           ;into DE
BDDA CDBB49   00940        CALL   49BBH
8DDD 28AF     00950        JR     Z,ERR4
8DDF 5F       00960        LD     E,A
BDE0 C9       00970        RET
8DE1 C5       00980 SUB2   PUSH   BC            ;Subroutine takes two ascii hex
8DE2 7B       00990        LD     A,E           ;digits in DE and returns the
8DE3 CDF48D   01000        CALL   SUB3          ;value of this number in A
8DE6 5F       01010        LD     E,A
8DE7 7A       01020        LD     A,D
BDEB CDF48D   01030        CALL   SUB3
8DEB 57       01040        LD     D,A
8DEC 7B       01050        LD     A,E
BDED 0610     01060        LD     B,16
8DEF 82       01070 AHAA   ADD    A,D
8DF0 10FD     01080        DJNZ   AHAA
8DF2 C1       01090        POP    BC
8DF3 C9       01100        RET
BDF4 E5       01110 SUB3   PUSH   HL            ;Subroutine converts A from sscii
8DF5 C5       01120        PUSH   BC            ;number to the number itself
8DF6 0610     01130        LD     B,16
8DF8 212B8C   01140        LD     HL,TABLE1
8DFB BE       01150 LOOP2  CP     (HL)
8DFC 23       01160        INC    HL
8DFD 2806     01170        JR     Z,OUT
8DFF 23       01180        INC    HL
8E00 10F9     01190        DJNZ   LOOP2
8E02 C38E8D   01200        JP     ERR4
8E05 7E       01210 OUT    LD     A,(HL)
8E06 C1       01220        POP    BC
8E07 E1       01230        POP    HL
8E08 C9       01240        RET
8E09 CDD48D   01250 SUB4   CALL   SUB1          ;Subroutine inputs four letters
8E0C CDE18D   01260        CALL   SUB2          ;from buffer and puts the
8E0F 67       01270        LD     H,A           ;corresponding number in HL
8E10 CDD48D   01280        CALL   SUB1
8E13 CDE18D   01290        CALL   SUB2
8E16 6F       01300        LD     L,A
BE17 C9       01310        RET
8E18 0600     01320 CONV2  LD     B,0           ;Subroutine takes number in A
8E1A D610     01330 LET    SUB    16            ;and converts it to ascii form
8E1C 3803     01340        JR     C,NOW         ;in HL
8E1E 04       01350        INC    B
8E1F 18F9     01360        JR     LET
8E21 C610     01370 NOW    ADD    A,16
8E23 CD2D8E   01380        CALL   CONV3
8E26 6F       01390        LD     L,A
8E27 78       01400        LD     A,B
8E28 CD2D8E   01410        CALL   CONV3
8E2B 67       01420        LD     H,A
8E2C C9       01430        RET
```

Program continues

```
8E2D E5        01440 CONV3   PUSH    HL              ;Subroutine converts hex in A to
8E2E C5        01450         PUSH    BC              ;ascii in A
8E2F 212C8C    01460         LD      HL,TABLE1+1
8E32 0610      01470         LD      B,16
8E34 4F        01480         LD      C,A
8E35 7E        01490 LOOP3   LD      A,(HL)
8E36 B9        01500         CP      C
8E37 2807      01510         JR      Z,OUT1
8E39 23        01520         INC     HL
8E3A 23        01530         INC     HL
8E3B 10F8      01540         DJNZ    LOOP3
8E3D C38E8D    01550         JP      ERR4
8E40 2B        01560 OUT1    DEC     HL
8E41 7E        01570         LD      A,(HL)
8E42 C1        01580         POP     BC
8E43 E1        01590         POP     HL
8E44 C9        01600         RET
               01610 ;
               01620 ;
               01630 ; X AND Y
               01640 ;
               01650 ;
8E45           01660 Y1      EQU     $
8B64           01670         ORG     Y
8B64 59        01680         DEFB    'Y'
8B65 458E      01690         DEFW    Y1
8E45           01700         ORG     Y1
8E45 21F09C    01710         LD      HL,9CF0H
8E48 11F95C    01720         LD      DE,BEGIN        ;Beginning of buffer
8E4B 010713    01730         LD      BC,SIZE         ;Temporary size of buffer
               01740                                 ;eventually 1000H larger
8E4E EDB0      01750         LDIR
8E50 21F95C    01760         LD      HL,BEGIN        ;Set top of buffer
8E53 46        01770 LOOP4   LD      B,(HL)
8E54 23        01780         INC     HL
8E55 7C        01790         LD      A,H
8E56 FE80      01800         CP      80H             ;Entire memory searched
```

Program continues

```
8E58 280F      01810          JR      Z,ERR5
8E5A 78        01820          LD      A,B
8E5B FEFF      01830          CP      0FFH
8E5D 20F4      01840          JR      NZ,LOOP4
8E5F 7E        01850          LD      A,(HL)
8E60 FEFF      01860          CP      0FFH
8E62 20EF      01870          JR      NZ,LOOP4
8E64 2B        01880          DEC     HL
8E65 221541    01890          LD      (4115H),HL      ;Top of buffer
8E68 C9        01900          RET
8E69 21F95C    01910 ERR5     LD      HL,BEGIN        ;Error, so make an empty buffer
8E6C 221541    01920          LD      (4115H),HL
8E6F 36FF      01930          LD      (HL),0FFH
8E71 23        01940          INC     HL
8E72 36FF      01950          LD      (HL),0FFH
8E74 C9        01960          RET
8E75           01970 X1       EQU     $
8B61           01980          ORG     X
8B61 58        01990          DEFB    'X'
8B62 758E      02000          DEFW    X1
8E75           02010          ORG     X1
8E75 2A1541    02020          LD      HL,(4115H)
8E78 23        02030          INC     HL
8E79 23        02040          INC     HL              ;End of buffer
8E7A 37        02050          SCF
8E7B 3F        02060          CCF
8E7C 11F95C    02070          LD      DE,BEGIN
8E7F ED52      02080          SBC     HL,DE           ;Size of buffer
8E81 E5        02090          PUSH    HL
8E82 E5        02100          PUSH    HL
8E83 21F95C    02110          LD      HL,BEGIN
8E86 11F09C    02120          LD      DE,9CF0H
8E89 C1        02130          POP     BC
8E8A EDB0      02140          LDIR                    ;Move into high memory
8E8C 21F09C    02150          LD      HL,9CF0H
8E8F D1        02160          POP     DE
8E90 19        02170          ADD     HL,DE           ;Top of high memory
8E91 E5        02180          PUSH    HL
8E92 7C        02190          LD      A,H
8E93 CD188E    02200          CALL    CONV2
8E96 7C        02210          LD      A,H
8E97 32C68E    02220          LD      (DATA),A
8E9A 7D        02230          LD      A,L
8E9B 32C78E    02240          LD      (DATA+1),A
8E9E E1        02250          POP     HL
8E9F 7D        02260          LD      A,L
8EA0 CD188E    02270          CALL    CONV2
8EA3 7C        02280          LD      A,H
8EA4 32C88E    02290          LD      (DATA+2),A
8EA7 7D        02300          LD      A,L
8EA8 32C98E    02310          LD      (DATA+3),A
8EAB 21B18E    02320          LD      HL,TEXT
8EAE C32B47    02330          JP      472BH
8EB1 28        02340 TEXT     DEFB    '('
8EB2 53        02350          DEFB    'S'
8EB3 54        02360          DEFB    'T'
8EB4 41        02370          DEFB    'A'
8EB5 52        02380          DEFB    'R'
8EB6 54        02390          DEFB    'T'
8EB7 3D        02400          DEFB    '='
8EB8 58        02410          DEFB    'X'
8EB9 27        02420          DEFB    ''''
8EBA 39        02430          DEFB    '9'
8EBB 43        02440          DEFB    'C'
8EBC 46        02450          DEFB    'F'
8EBD 30        02460          DEFB    '0'
8EBE 27        02470          DEFB    ''''
8EBF 2C        02480          DEFB    ','
8EC0 45        02490          DEFB    'E'
8EC1 4E        02500          DEFB    'N'
8EC2 44        02510          DEFB    'D'
8EC3 3D        02520          DEFB    '='
8EC4 58        02530          DEFB    'X'
8EC5 27        02540          DEFB    ''''
8EC6 2A        02550 DATA     DEFB    '*'
8EC7 2A        02560          DEFB    '*'
8EC8 2A        02570          DEFB    '*'
8EC9 2A        02580          DEFB    '*'
8ECA 27        02590          DEFB    ''''
8ECB A9        02600          DEFB    ')'+128
               02610 ;
               02620 ;
               02630 ; J
```

Program continues

ACCEL2: Compiler for TRS-80 Disk BASIC. Compiles selected subset to Z80 machine code in all four variable types, compact 1K run-time component controls interpreter to streamline all other statements and functions. Technique minimises code expansion without impairing huge speedups for true double optimisation. Six diagnostic messages. Local/Global options increase compatibility with subject programs. Output save to Disk, instructions for self-contained SYSTEM tape. Professionals note: No royalties on the derived code! ACCEL2 brings your BASIC programs **alive**. It's like having a 100 mhz clock! **$88.95**

Developed by Southern Software in England, now available in US from...

ALLEN GELDER SOFTWARE ✓79
Box 11721 Main Post Office
San Francisco, CA 94101

TRS-80 tm Radio Shack/Tandy Corp.

TEXAS COMPUTER SYSTEMS

Radio Shack

Authorized Sales Center, OFFERS

LOWEST PRICES on

TRS-80

COMPUTERS

For the BEST prices on ALL TRS-80 computers, CALL our TOLL FREE NUMBER 1-800-351-1473. All Radio Shack® computers are discounted 10%, 15% up to 20%! CALL for the latest prices on the items you need, or get advice from our consultant about your specific needs. CALL for prices on the Model I, II and the new Model III, Color Computer and Pocket Computer.

SAVE up to **50%** on accessories (non-Radio Shack®). Need more disk space? Ask about single/**DOUBLE DENSITY** controller for the Model I. 300k in a 2-disk system. 5 minute installation w/no modifications. Copies your single density data to **DOUBLE** for complete compatability. Less than $200.

40 track disk drives $359. 16k memory add on only $58 w/instructions. Specify computer or expansion interface. CALL for information on Programs available.

- ★ UPS prepaid insured delivery—**FREE** except some large items
- ★ No taxes on out-of-state shipments. Texas res. Add 5%
- ★ All merchandise is new, checked and guaranteed by manufacturer
- ★ Payment: Money Order, Cashier's Check, Certified Check. Personal Checks require 3 weeks to clear. VISA, MASTERCHARGE—Add 3%
- ★ Prices subject to change at any time
- ★ Delivery of merchandise is subject to availability

✓25

TCS, 106 East 10th, Brady, TX. 76825

An Authorized **RADIO SHACK®** Sales Center F701

TOLL FREE Order Number 1-800-351-1473

Texas Residents 915-597-0673

CalData Systems Presents ✓294

WordMagic II

Complete WORD PROCESSING designed specifically for The Radio Shack TRS-80 Model II Computer

WordMagic II™ is a Word Processor designed specifically for the Radio Shack TRS 80 Model II Computer.*

FEATURES INCLUDE:

- •Mailing List/Labels Generation
- •Automatic Merging of Mailing Data with Text Files to created "PERSONALIZED" Form Letters
- •Automatic wrap-around in text entry
- •Margination, Paging, Complete Cursor Movement
- •Complete Editing Commands—Insertion, Global Substitution, Overwrite, etc.
- •Centering, Smooth Right, Left Justify
- •Teble of Contents Generation
- •Automatic Page Numbering
- •Variable Form Lengths
- •Underlining
- •Line Numbering

TRS-80 is a trademark of the Radio Shack Division of Tandy Corporation

Requires 64K Model II, TRSDOS 2 BASIC (not provided with WordMagic)

Introductory Pricing: $195.00 Manual $20 (Cal. Res. add tax)

CalData Systems P.O. Box 178446
San Diego, CA 92117 (714) 272-2661

---

250 PROGRAMS!

65 Page Software Catalog

That's how many are included in our Fall/Winter catalog, and are in stock and ready to ship. We represent all of the major TRS-80 Software Vendors (Instant Software, The Bottom Shelf, Softside, Small Business Systems Group, and many more) — Over 50 Vendors represented in our product line. We believe we have the most complete selection of software for the TRS-80 — all priced at manufacturer's prices from $7.95 to $99.95. Business, home, games, education, programmer's utilities. COMPLETE program descriptions.

SEND FOR OUR FREE ✓470
65 PAGE CATALOG TODAY!

MICRO COMPUTER SYSTEMS
3104 EAST SHADOWLAWN, N.E.
ATLANTA, GA. 30305

CASIO

Calculator Watch

4 Function Calculator
Time - Calendar Display
1/100 sec. stopwatch
Dual Time

$42.95

X-10 SYSTEM

Command Console	31.95
Ultrasonic Console	33.95
Cordless Controller	16.95
TIMER NEW!	59.95
Lamp, Appliance Module	13.49
Wall Switch	14.95

TELEPHONE ANSWERING

Code-A-Phone	Record a Call
1000..... 97.95	80A..... 199.95
1750..... 209.95	90A..... 279.95

PRINTERS

EPSON MX-80	499.00
CENTRONICS 737	799.00

SYNAPSE VIDEO ✓268
P.O. BOX 962
New York, N.Y. 10009

```
                02640 ;
                02650 ;
8ECC            02660 J1        EQU     $
8B67            02670           ORG     J
8B67 4A         02680           DEFB    'J'
8B68 CC8E       02690           DEFW    J1
8ECC            02700           ORG     J1
8ECC CDBB49     02710           CALL    49BBH           ;Next letter
8ECF CA8E8D     02720           JP      Z,ERR4
8ED2 FE55       02730           CP      'U'
8ED4 2004       02740           JR      NZ,FANCY
8ED6 CD098E     02750           CALL    SUB4
8ED9 E9         02760           JP      (HL)
8EDA CDAF88     02770 FANCY     CALL    RETN
8EDD 210040     02780           LD      HL,4000H        ;Save bottom of EDTASM
8EE0 11F099     02790           LD      DE,99F0H
8EE3 010003     02800           LD      BC,300H
8EE6 EDB0       02810           LDIR
8EE8 CD098E     02820           CALL    SUB4
8EEB 31EF99     02830           LD      SP,99EFH        ;Put stack elsewhere
8EEE EB         02840           EX      DE,HL
8EEF 21018F     02850           LD      HL,CLEAN        ;Will return here after jump
8EF2 E5         02860           PUSH    HL
8EF3 D5         02870           PUSH    DE
8EF4 21368F     02880           LD      HL,BASIC        ;Put BASIC fixed ROM in position
8EF7 110040     02890           LD      DE,4000H
8EFA 010003     02900           LD      BC,300H
8EFD EDB0       02910           LDIR
8EFF E1         02920           POP     HL
8F00 E9         02930           JP      (HL)
8F01 F3         02940 CLEAN     DI                      ;Return bottom of EDTASM to 4000H
8F02 21F099     02950           LD      HL,99F0H
8F05 110040     02960           LD      DE,4000H
8F08 010003     02970           LD      BC,300H
8F0B EDB0       02980           LDIR
8F0D C3DA46     02990           JP      46DAH
                03000 ;
                03010 ;
                03020 ; DEBOUNCE ROUTINE FOR BASIC ROM
                03030 ;
                03040 ;
8F10            03050 BASDEB    EQU     $
8F10 213640     03060           LD      HL,16438
8F13 010138     03070           LD      BC,14337
8F16 1600       03080           LD      D,0
8F18 0A         03090 LI        LD      A,(BC)
8F19 5F         03100           LD      E,A
8F1A AE         03110           XDR     (HL)
8F1B 73         03120           LD      (HL),E
8F1C A3         03130           AND     E
8F1D 2008       03140           JR      NZ,NI
8F1F 14         03150           INC     D
8F20 2C         03160           INC     L
8F21 CB01       03170           RLC     C
8F23 F2188F     03180           JP      P,LI
8F26 C9         03190           RET
8F27 5F         03200 NI        LD      E,A
8F28 C5         03210           PUSH    BC
8F29 01DF04     03220           LD      BC,1247
8F2C CD6000     03230           CALL    96
8F2F C1         03240           POP     BC
8F30 0A         03250           LD      A,(BC)
8F31 A3         03260           AND     E
8F32 C8         03270           RET     Z
8F33 C3FB03     03280           JP      1019
8F36            03290 BASIC     EQU     $
9236            03300 CDNT3     EQU     BASIC+300H
                03310 ;
                03320 ;
                03330 ; TEMPORARY CODE TO PLACE BASIC IN POSITION
                03340 ;
                03350 ;
A000            03360           ORG     0A000H
A000 21108F     03370           LD      HL,BASDEB
A003 221640     03380           LD      (16406),HL
A006 F3         03390           DI
A007 210040     03400           LD      HL,4000H
A00A 11368F     03410           LD      DE,BASIC
A00D 010003     03420           LD      BC,300H
A010 EDB0       03430           LDIR
A012 C30000     03440           JP      0
0000            03450           END
00000 TOTAL ERRORS
```

NEW FROM MICRO-SYSTEMS!!!

Micro-Systems Software Inc. now offers you two new, and powerful disk backup utilities, superback and multiback.

Superback is the backup program you've always dreamed of, but no-one ever sold. Superback will backup any 5 1/4" diskette that will work in your TRS-80*. Anything that will boot-up, we will backup. Specifically, this program will scan any diskette created by either the 1771 or 1791 disk controller chips. After scanning the disk, the program will then proceed to back it up with the same format that it currently has.

Multiback is our new multiple backup program. It is specifically designed for smaller software houses, like ourselves, who don't have the giant diskette copying machines. It will make three backups of a single disk simultaneously. Simply insert the master diskette in drive zero, and blanks in drives one, two and three, press enter and this program will copy the first disk onto the other three. A real time-and-effort saver!

Superback and multiback both retail for **$19.95**. To order, contact us at the address below.

master charge | VISA

MICRO-SYSTEMS SOFTWARE-INC

Specializing in the Tandy Line

✓384

(305) 983-3390

5846 Funston Street
Hollywood, FL 33023

*TRS - 80 is a trademark of Tandy Corp.
- Doubler is a trademark of Percom Data Corp.

Now available in Spanish

STOP PLAYING GAMES

TRS-80 (Level II)
APPLE
OTHERS

- Calculate odds on HORSE RACES with ANY COMPUTER using **BASIC**.
- SCIENTIFICALLY DERIVED SYSTEM really works. TV Station WLKY of Louisville, Kentucky used this sytem to predict the odds of the 1980 Kentucky Derby. See the Wall Street Journal (June 6, 1980) article on Horse-Handicapping. This system was written and used by computer experts and is now being made available to home computer owners. This method is based on storing data from a large number of races on a high speed, large scale computer. 23 factors taken from the "Daily Racing Form" were then analyzed by the computer to see how they influenced race results. From these 23 factors, ten were found to be the most vital in determining winners. NUMERICAL PROBABILITIES of each of these 10 factors were then computed and this forms the basis of this REVOLUTIONARY NEW PROGRAM.
- SIMPLE TO USE. Obtain "Daily Racing Form" the day before the races and answer the 10 questions about each horse. Run the program and your computer will print out the odds for all horses in each race. COMPUTER POWER gives you the advantage!
- YOU GET: 1) TRS-80 (Level II) or Apple Cassette
  2) Listing of BASIC program for use with any computer
  3) Instructions on how to get the needed data from the "Daily Racing Form"
  4) Tips on using the odds generated by the program
  5) Sample form to simplify entering data for each race

MAIL COUPON OR CALL TODAY

3G COMPANY, INC. DEPT. M-80 **(503) 357-9889**
RT. 3, BOX 28A, GASTON, OR 97119

Yes, I want to use my computer for FUN and PROFIT. Please send me _____ programs at **$19.95 each.**

I need a ☐ TRS-80 Cassette or ☐ Apple Cassette

Enclosed is: ☐ check or money order ☐ Master Charge ☐ Visa

Card No _____ Exp. date _____

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

START USING YOUR COMPUTER FOR FUN and PROFIT!

Complete LNW Expansion Interfaces

The LNW System Expansion offers one of the best alternatives to the Radio Shack interface, and now with a complete kit from COMPUTEX it's even better. We studied the I.C. market for three months and averaged the cost of procurring components for the LNW board. We found that by shopping for the best prices from over 10 vendors, the LNW board could be assembled for an average parts cost of $ 253.00 not including shipping cost. COMPUTEX saves you time and money by offering a complete LNW system expansion kit for $ 249.00 (less RAM and Cassette Relay). We even include all I.C. Sockets. Not only is the LNW/COMPUTEX expansion interface better electrically, we've made it the best TRS80(tm*) expansion interface by designing and building a custom cabinet for it. *TRS80 is a trademark of Tandy Corp.

The CPT Cabinets for the LNW interfaces are made of quality birch wood, custom finished in a light walnut color then trimmed around the front by aluminum molding. Two cabinets are available.

The CPT1000 cabinet will hold the LNW Board, and power supplies for both keyboard and the LNW system expansion. Measurements 15" wide x 13¼" deep x 5¾" tall. . . . **$ 89.95**

The CPT2000 cabinet has all of the features of the CPT1000 but will hold up to two disk drives, power supplies, and even has a cut out for a muffin fan. The CPT2000 has a removeable front panel that comes with cutouts for 1 or 2 disk drives or with no cutouts. The CPT2000 measures 15¾" wide x 13¾" deep x 3½" tall. . . . . **$ 99.95**

LNW System Expansion Kit	$ 249.00
(Assembled)	349.00
CPT1000 Cabinet	89.95
CPT2000 Cabinet	99.95
LNW System Expansion Board	69.00
T1 (Radio Shack) Transformer for LNW	21.95
Keyboard to FI/F Cable	19.95
Muffin Fan for CPT2000	14.95

(Individual components avadable also)

Disk Drives

COMPUTEX reviewed all major disk drives available on the market prior to becoming a dealer for anyone. The drive we selected to market is the Tandon TM 100 Series. Compare their specifications and features and we think you'll agree that the Tandon TM 100 Series of Disk Drives are the best available.

- Tandon is the leading designer and supplier of read/write heads for most other disk drive manufacturers
- Track to Track access time of 5 milliseconds
- No head load time required, most others take 35 M.S.
- Read / write head guaranteed for 20,000 hours
- Quieter than most other disk drives
- First successful manufacturer of double headed drives

Model	Description	Base Price	With Supply Case
TM100-1	40 Track Single Headed	$225.00	$299.95
TM100-2	40 Track Double Headed	325.00	399.95
TM100-3	80 Track Single Headed	375.00	449.00
TM100-4	80 Track Double Headed	475.00	549.00

All above drives will operate single or double density

For those that still insist on MPI and Shugart

	Base Unit	With Supply Case
MPIB51-40 Track Single Headed	$275.00	349.00
MPIB52-40 Track Double Sided	375.00	449.00
Shugart SA400-35 Track Single Sided	255.00	329.00

Computex carries or can supply most any TRS80 System or peripherals. (Call for quotes)

VARIOUS OTHER SPECIALS!

Novation / Cat Modem	$ 179.00
Verbatim / Scotch diskettes (Box 10)	24.95
16 K RAM Chips	8 / 54.95
Radio Shack Systems	
Level II - 16K RAM	$ 700.00
Level II - 4 K RAM	595.00
Model II - 64K System	3,500.00
OK Expansion Interface	$ 259.00
16K Expansion Interface (our RAM)	339.00
32K Expansion Interface (our RAM)	419.00
Software	
Newdos 80	$ 149.00
Dosplus	99.95
Electric Pencil(model I disk)	$ 150.00
Electric Pencil(model I cassette)	100.00

Centronics 737

For the first time ever, true letter quality printing for under $800. Compare quality, features and our price. We think you'll agree that the 737 printer from COMPUTEX is unbeatable.

FEATURES

Fan fold, letterhead and roll feed paper
TRUE UNDERLINING CAPABILITIES
Subscript and superscript printing
Fast 80 CPS (proportionall and 50 CPS (monospaced)
True descending lower case
Right margin justification
Optional foreign character sets
Nx9 dot matrix or 7x8 dot matrix
Expanded print-10 CPI and 16.7 CPI
Bidirectional stepper motor
80 or 132 column printing
Best of all is the price. . . . . **$ 750.00**

All products sold by COMPUTEX are 100% guaranteed for 90 days. A 1 year 100% guarantee is available on all of our hardware for an additional 10% of the items purchase price.

VISA/Master Card accepted(add 4% to total) ALL ORDERS SHIPPED WITHIN 6 DAYS OF ORDER SHIPPING —UPS insured(call for rate) Personal checks held 2 weeks prior to shipping.

C.O.D.'s accepted(may require 10% down)

✓415 **Computex**

17710 Heritage Ct., Webster, Tx. 77598 (713) 332-4359

The program capitalizes automatically unless you add lowercase. The first letter of each sentence and the word I are always capitalized. Other characters will be printed in lowercase unless they are entered with SHIFT. The computer begins a sentence when it recognizes a period or question mark followed by two spaces.

TAB indicates the beginning of a new paragraph. TAB should not be used to jump from column to column if you intend to print with TYPE.

M2 and TYPE can be used to create titles and section headings. Enter a title by typing #, the title, and another #. The title will be centered. Since most titles consist of capital letters, the computer will print UNSHIFTED letters as capitals and SHIFTED letters in lowercase unless you add the lowercase modification.

The characters ## inserted into the buffer alone are used to print a blank line.

For housekeeping reasons, the computer will not print initial spaces on a line. It can be made to jump to the middle of a line by entering a tab and then continuing with the appropriate number of spaces.

Several printing parameters can be changed using POKE. A detailed list of printing parameters is given in Table 1. The number of characters on a line, the size of the left margin, and the paragraph indentation can be chosen at will. The computer can be made to single space or double space, it will add an additional line between paragraphs if desired, and it can be made to print titles using wide characters if your printer has that ability.

The program will work with printers as narrow as Radio Shack's 32 column wide Quick Printer II. Obviously, a wider printer is preferable.

Making the Modification

To modify the assembler, follow the instructions at the end of this article. The modification proceeds in stages. First EDTASM is placed on disk using a famous trick. Next it is given the ability to assemble directly into memory; this step requires a tedious entry of code by hand. From then on, the assembler is used to modify itself. Modification becomes easier and easier as the assembler changes.

There are two versions of EDTASM, 1.1 and 1.2. If you own version 1.2, pay close attention to the comments which begin sections three, four, and five; they describe a small number of changes you must make in the listed code.

Other comment lines interspersed throughout the code describe changes you may wish to make in startup mode, printer

continuous to page 250

Program Listing 6. EDTASM 6

```
                00100 ; INSTRUCTIONS(A):  Run EDTASM5.  The code below converts the assembler
                00110 ; to lower case for readers who have installed Radio Shack's lower
                00120 ; case modification.  This code will work with most other lower
                00130 ; case modifications; rewrite it if necessary.  If you do not have
                00140 ; a lower case modification, go to the next set of instructions.
                00150 ;      When the code below is in place, the keyboard will work
                00160 ; exactly as it works under Radio Shack's BASIC software.  EDTASM
                00170 ; will power up in upper case mode.  To convert back and worth
                00180 ; between this mode and regular typewriter mode, press SHIFT-ZERO.
                00190 ;      The automatic capitolization feature of TYPE is intended
                00200 ; for those without lowercase.  If you make the lower case modifi-
                00210 ; cation, TYPE will print exactly what is on the screen.
                00220 ;
                00230 ;
                00240 ; LOWER CASE
                00250 ;
                00260 ;
9236            00270 CONT3    EQU     9236H
8C4F            00280 HERE     EQU     8C4FH
8B90            00290 OHH      EQU     8B90H
734A            00300          ORG     734AH           ;Keyboard accept all letters
734A 80         00310          DEFB    80H
8B90            00320          ORG     OHH             ;Cancel "Method" case control
8B90 00         00330          DEFB    0
8B91 0000       00340          DEFW    0
717B            00350          ORG     717BH           ;Modify screen routine to print
717B 18         00360          DEFB    18H             ;exactly what is received
731A            00370          ORG     731AH           ;Modify keyboard
731A C3         00380          DEFB    0C3H
731B 3692       00390          DEFW    CONT3
9236            00400          ORG     CONT3
9236 B7         00410          OR      A               ;Code replaced by JP
9237 CA1746     00420          JP      Z,4617H
923A F5         00430          PUSH    AF
923B 3A6592     00440          LD      A,(LOWER)       ;Keyboard mode
923E FE01       00450          CP      1
9240 280C       00460          JR      Z,REGULA
9242 F1         00470          POP     AF
9243 FE61       00480          CP      61H             ;Shifted letter
9245 3806       00490          JR      C,NOS
9247 FE7B       00500          CP      7BH
9249 3002       00510          JR      NC,NOS
924B D620       00520          SUB     20H
924D C9         00530 NOS      RET
924E F1         00540 REGULA   POP     AF
924F FE7B       00550          CP      7BH             ;Shifted letter
9251 30FA       00560          JR      NC,NOS
9253 FE61       00570          CP      61H
9255 3803       00580          JR      C,NOS1
9257 D620       00590          SUB     20H
```

Program continues

Reader Service—see page 274

```
9259 C9        00600          RET
925A FE5B      00610 NOS1     CP      5BH             ;Unshifted letter
925C 30EF      00620          JR      NC,NOS
925E FE41      00630          CP      41H
9260 38EB      00640          JR      C,NOS
9262 C620      00650          ADD     A,20H
9264 C9        00660          RET
9265 00        00670 LOWER    DEFB    0               ;Keyboard mode
9266           00680 KEY      EQU     $
8C5C           00690          ORG     HERE+0DH        ;Immediately after debounce
8C5C 6692      00700          DEFW    KEY             ;see if shift-zero is pressed
9266           00710          ORG     KEY             ;if so, change keyboard
9266 F5        00720          PUSH    AF
9267 7A        00730          LD      A,D             ;Zero key
9268 FE04      00740          CP      4
926A 2016      00750          JR      NZ,NOC
926C 7B        00760          LD      A,E
926D FE01      00770          CP      1
926F 2011      00780          JR      NZ,NOC
9271 3A8038    00790          LD      A,(3880H)       ;Shift
9274 FE01      00800          CP      1
9276 200A      00810          JR      NZ,NOC
9278 3A6592    00820          LD      A,(LOWER)
927B EE01      00830          XOR     1
927D 326592    00840          LD      (LOWER),A
9280 F1        00850          POP     AF
9281 C9        00860          RET
9282 F1        00870 NOC      POP     AF
9283 C30A44    00880          JP      440AH
               00890 ;
               00900 ;
               00910 ; INSTRUCTIONS(B):  The code below forms a new printer driver.
               00920 ; When the driver is in place, you may use both serial and parallel
               00930 ; printers.  To convert to the serial printer, POKE 9327,01.
               00940 ; To convert back to the psrallel printer, POKE 9327,00.
               00950 ;      The IN and OUT commands in the program refer to the serial
               00960 ; printer.  If you do not have a serial printer, keep these
               00980 ;      Since printers differ, you may have to make some changes
               00990 ; in the code.  Read the comments sprinkled throughout the text.
               01000 ;      When using this driver, you may press the BREAK key at any
               01010 ; time to stop printing and return to EDTASM.  The printer can
               01020 ; type any desired number of lines per page.  As written, the
               01030 ; printer will type continuously.  To change the number of lines
               01040 ; per page, POKE 932A with the new number of lines (in hex).
               01050 ; The printer will then print this number of lines, skip to the
               01060 ; next page, and resume printing there.  It is assumed that the
               01070 ; total number of possible lines on a page is 66; this number too
               01080 ; can be changed by poking 932B with the new number (in hex).
               01090 ; To reset the line counter at 0 (top of form), POKE 9329 with 00.
               01100 ; Finally, it is possible to pause at the end of each page if
               01110 ; you are typing on single sheets.  To do so, POKE 932C with 01.
               01120 ; Continuous printing without these pauses can be resumed by
               01130 ; poking 932C with 00.
               01140 ;
               01150 ;
               01160 ; SERIAL PRINTER DRIVER
               01170 ;
               01180 ;
9286           01190 PRINT    EQU     CONT3+50H
72AA           01200          ORG     72AAH
72AA C3        01210          DEFB    0C3H
72AB 8692      01220          DEFW    PRINT
9286           01230          ORG     PRINT
9286 3A2793    01240          LD      A,(PTYPE)
9289 B7        01250          OR      A
928A F5        01260          PUSH    AF
928B 2816      01270          JR      Z,AROUND
928D 3A2893    01280 SERIAL   LD      A,(INIT)        ;Already initialized?
9290 FE01      01290          CP      1
9292 280F      01300          JR      Z,AROUND
9294 3E01      01310          LD      A,1             ;If not, initialize
9296 322893    01320          LD      (INIT),A
9299 D3E8      01330          OUT     (0E8H),A        ;Reset RS-232-C
               01340 ; The next instruction sets the transmission rate at 1200 baud.
               01350 ; Use 55H for 300 baud and 66H for 600 baud.  Consult the RS-232
               01360 ; manual for other values.
929B 3E77      01370          LD      A,77H           ;1200 baud
929D D3E9      01380          OUT     (0E9H),A
               01390 ; The next instruction selects even parity, one stop bit, and
               01400 ; word length seven.  Consult the RS-232 manual for other values.
929F 3EA4      01410          LD      A,0A4H
92A1 D3EA      01420          OUT     (0EAH),A
```

Program continues

```
92A3 F1       01430 AROUND  POP     AF
92A4 CDFA92   01440         CALL    OUTPUT
92A7 FE0D     01450         CP      13              ;Carriage return
92A9 C0       01460         RET     NZ
              01470 ; Replace the next command with NOP, NOP if your printer does
              01480 ; not issue automatic line feeds after carriage returns.
92AA 1805     01490         JR      AR3
92AC 0E0A     01500         LD      C,10            ;Line feed
92AE CDFA92   01510         CALL    OUTPUT
92B1 3A2A93   01520 AR3     LD      A,(PAGE)
92B4 47       01530         LD      B,A
92B5 3A2993   01540         LD      A,(LINE1)
92B8 3C       01550         INC     A
92B9 322993   01560         LD      (LINE1),A
92BC B8       01570         CP      B
92BD C0       01580         RET     NZ
92BE 3A2B93   01590         LD      A,(TOTAL)
92C1 90       01600         SUB     B
92C2 B7       01610         OR      A
92C3 280F     01620         JR      Z,AR4
92C5 47       01630         LD      B,A
92C6 0E0D     01640 AR5     LD      C,13
92C8 CDFA92   01650         CALL    OUTPUT
              01660 ; Replace the next command with NOP, NOP if your printer does
              01670 ; not issue automatic line feeds after carriage returns.
92CB 1805     01680         JR      AR6
92CD 0E0A     01690         LD      C,10
92CF CDFA92   01700         CALL    OUTPUT
92D2 10F2     01710 AR6     DJNZ    AR5
92D4 3E00     01720 AR4     LD      A,0
92D6 322993   01730         LD      (LINE1),A
92D9 3A2C93   01740         LD      A,(PAUSE)
92DC FE01     01750         CP      1
92DE C0       01760         RET     NZ
92DF 3A4038   01770 AR7     LD      A,(3840H)
92E2 FE04     01780         CP      4               ;Break
92E4 CADA46   01790         JP      Z,46DAH
92E7 FE01     01800         CP      1               ;Enter
92E9 C8       01810         RET     Z
92EA FE80     01820         CP      128             ;Space
92EC C8       01830         RET     Z
92ED 18F0     01840         JR      AR7
92EF F5       01850 BREAK   PUSH    AF
92F0 3A4038   01860         LD      A,(3840H)
92F3 FE04     01870         CP      4
92F5 CADA46   01880         JP      Z,46DAH
92F8 F1       01890         POP     AF
92F9 C9       01900         RET
92FA 3A2793   01910 OUTPUT  LD      A,(PTYPE)
92FD B7       01920         OR      A
92FE 2816     01930         JR      Z,PARALL
9300 DBEA     01940         IN      A,(0EAH)        ;Transmitter holding register
9302 CB77     01950         BIT     6,A
9304 CDEF92   01960         CALL    BREAK
9307 28F1     01970         JR      Z,OUTPUT
              01980 ; The next four instructions provide a handshake with the printer.
              01990 ; If you do not have handshake capability, replace them with
              02000 ; nine NOP's.  You may want to look at a different status bit;
              02010 ; consult the manual if so.
9309 DBE8     02020 AR2     IN      A,(0E8H)        ;Printer ready?
930B CB77     02030         BIT     6,A
930D CDEF92   02040         CALL    BREAK
9310 20F7     02050         JR      NZ,AR2
9312 79       02060         LD      A,C
9313 D3EB     02070         OUT     (0EBH),A
9315 C9       02080         RET
9316 3AE837   02090 PARALL  LD      A,(37E8H)
9319 E6F0     02100         AND     0F0H
931B FE30     02110         CP      30H
931D CDEF92   02120         CALL    BREAK
9320 20F4     02130         JR      NZ,PARALL
9322 79       02140         LD      A,C
9323 32E837   02150         LD      (37E8H),A
9326 C9       02160         RET
              02170 ; Replace the next value by zero if you do not have a serial
              02180 ; printer or if you want to power up in parallel printer mode.
9327 01       02190 PTYPE   DEFB    01
9328 00       02200 INIT    DEFB    0
9329 00       02210 LINE1   DEFB    0
932A 42       02220 PAGE    DEFB    66
932B 42       02230 TOTAL   DEFB    66
932C 00       02240 PAUSE   DEFB    0
932D          02250 CONT4   EQU     CONT3+0F7H
```

Program continues

```
                02260 ;
                02270 ;
                02280 ; INSTRUCTIONS(C):  It is possible to use both cassettes with
                02290 ; EDTASN.  When the assembler powers up, the first cassette
                02300 ; will be selected.  Issue POKE 37E4,01 to select the second
                02310 ; cassette and POKE 37E4,00 to reselect the first.  There will be
                02320 ; an audible click from the cassette select relay when the assembler
                02330 ; powers up.  If you seldom use the cassettes and the click
                02340 ; annoys you, poke 00 into locations 73BA, 73BB, and 73BC now.
                02350 ; It will still be possible to use the cassettes, but you
                02360 ; must then POKE 37E4 with the appropriate value before doing so.
                02370 ;
                02380 ;
                02390 ; INSTRUCTIONS(D):  The assembler can assemble directly into
                02400 ; any memory location between A000H and BFFFH.  If you have 48K
                02410 ; of memory, the last 16K have been protected from EDTASN.  But
                02420 ; if you went to assemble directly into this area too (and so
                02430 ; anywhere from A000H to FFFFH), then POKE 8AB9,FF now.
                02440 ;
                02450 ;
                02460 ; INSTRUCTIONS(E):  Finally, POKE 75E1,32 now to convert the
                02470 ; startup message from "Version 1.1" to "Version 2.1" (respectively
                02480 ; "Vsrsion 1.2" to "Version 2.2").
                02490 ;       Assemble the above text to memory.  If you are going to
                02500 ; add the TYPE command, issue
                02510 ; DUMP EDTASN6/CMD (START=X'7000',END=X'932D',TRA=X'8A00').
                02520 ;       If you are going to stop here, POKE 8AB1,00 and POKE 8AB2,00
                02530 ; to protect the assembler from assembling code on top of itself.
                02540 ; Then POKE 7397,7F to enlarge the text buffer, and POKE 8E4D,23
                02550 ; to modify SIZE accordingly.  Issue the command M1 if you want
                02560 ; EDTASN to power up in keyboard entry mode one (and M0 or M2
                02570 ; for other entry modes).  Finally, if you added the lower case
                02580 ; modification, put the assembler in the mode in which you wish
                02590 ; it to power up.  Issue the command
                02600 ; DUMP EDTASN6/CMD (START=X'7000',END=X'932D',TRA=X'8A00').
0000            02610       END
00000 TOTAL ERRORS
```

Program Listing 7. EDTASM 7

```
                00100 ; INSTRUCTIONS:  Run EDTASN6.  If you stopped after the last
                00110 ; modifications and later decided to add the TYPE command, issue
                00120 ; the commands POKE 8AB1,18 and POKE 8AB2,B0 to allow assembly
                00130 ; anywhere in memory, issue POKE 4114,6F and POKE 7392,6F to
                00140 ; restrict the size of the edit buffer, and issue POKE 8E4D,13
                00150 ; to modify SIZE accordingly.
                00160 ;       Enter the code below and assemble it to memory.  Then enter
                00170 ; DUMP EDTASN7/CMD (START=X'7000',END=X'9501',TRA=X'8A00').
                00180 ;       Do not use the TYPE command of EDTASN7; it will crash the
                00190 ; system.
                00200 ;       Since printers differ, you may have to change a few
                00210 ; instructions below.  These instructions are preceded by
                00220 ; comment lines; read all such lines carefully.
                00230 ;
                00240 ;
                00250 ; CODE TO DETECT TYPE COMMAND
                00260 ;
                00270 ;
932D            00280 CONT4   EQU     932DH
8BAF            00290 RETN    EQU     8BAFH
8D8A            00300 LETTER  EQU     8D8AH
8D8E            00310 ERR4    EQU     8D8EH
8B20            00320 CONT    EQU     8B20H
9286            00330 PRINT   EQU     9286H
760C            00340         ORG     760CH
760C 2D93       00350         DEFW    CONT4
932D            00360         ORG     CONT4
932D E5         00370         PUSH    HL
932E D5         00380         PUSH    DE
932F C5         00390         PUSH    BC
9330 F5         00400         PUSH    AF
9331 CDBB49     00410         CALL    49BBH           ;Next letter
9334 FE59       00420         CP      'Y'
9336 280A       00430         JR      Z,TYPE1
9338 CDAF8B     00440         CALL    RETN            ;Replace letter
933B F1         00450         POP     AF
933C C1         00460         POP     BC
933D D1         00470         POP     DE
```

Program continues

```
933E E1        00480          POP     HL
933F C31C4C    00490          JP      4C1CH           ;Regular type command
9342 CD8A8D    00500 TYPE1    CALL    LETTER          ;Next letter
9345 FE50      00510          CP      'P'
9347 C28E8D    00520          JP      NZ,ERR4
934A CD8A8D    00530          CALL    LETTER
934D FE45      00540          CP      'E'
934F C28E8D    00550          JP      NZ,ERR4
9352 218F93    00560          LD      HL,TYPEA        ;Initialize type command
9355 362E      00570          LD      (HL),'.'
9357 23        00580          INC     HL
9358 3620      00590          LD      (HL),' '
935A 23        00600          INC     HL
935B 3620      00610          LD      (HL),' '
935D 23        00620          INC     HL
935E 3600      00630          LD      (HL),0
9360 23        00640          INC     HL
9361 3600      00650          LD      (HL),0
9363 219A93    00660          LD      HL,BUFF
9366 229493    00670          LD      (TYPEA+5),HL
9369 3E00      00680          LD      A,0
936B 329693    00690          LD      (TYPEA+7),A
936E 211A94    00700          LD      HL,BUFFT
9371 229793    00710          LD      (TYPEA+8),HL
9374 213947    00720          LD      HL,4739H        ;Intercept text output
9377 36C3      00730          LD      (HL),0C3H
9379 21BC94    00740          LD      HL,CHANGE       ;Intercept printer output
937C 22AB45    00750          LD      (45ABH),HL
937F 21AE94    00760          LD      HL,NOTAB
9382 223A47    00770          LD      (473AH),HL
9385 F1        00780          POP     AF
9386 C1        00790          POP     BC
9387 D1        00800          POP     DE
9388 E1        00810          POP     HL
9389 CD1C4C    00820          CALL    4C1CH
938C C36C95    00830          JP      STUFF
938F 2E        00840 TYPEA    DEFB    '.'     ;Last three characters inserted
```

Program continues

```
9390 20        00850          DEFB    ' '              ;into buffer
9391 20        00860          DEFB    ' '
9392 00        00870          DEFB    0                ;One if title being entered
9393 00        00880          DEFB    0                ;Number of letters in buffer
9394 9A93      00890          DEFW    BUFF             ;Next buffer spot
9396 00        00900          DEFB    0                ;Number of letters in title buffer
9397 1A94      00910          DEFW    BUFFT            ;Next title buffer spot
               00920 ; The next address contains the number of characters on a
               00930 ; printed line.  This number should be 31 for the Radio Shack
               00940 ; Quick Printer II and about 60 for a regular printer.  It can
               00950 ; be as large as 127.
9399 3C        00960 BUFFL    DEFB    60
939A           00970 BUFF     EQU     $
941A           00980 BUFFT    EQU     BUFF+80H
949A           00990 INIT2    EQU     BUFF+100H
               01000 ;
               01010 ;
               01020 ; ADDITIONAL INITIALIZATION BEFORE EDTASM PROMPT
               01030 ;
               01040 ;
8B43           01050          ORG     CONT+23H
8B43 9A94      01060          DEFW    INIT2
949A           01070          ORG     INIT2
949A 213947    01080          LD      HL,4739H
949D 36F5      01090          LD      (HL),0F5H
949F 23        01100          INC     HL
94A0 36E5      01110          LD      (HL),0E5H
94A2 23        01120          INC     HL
94A3 3621      01130          LD      (HL),21H
94A5 218692    01140          LD      HL,PRINT
94A8 22AB45    01150          LD      (45ABH),HL
94AB C3DD46    01160          JP      46DDH
               01170 ;
               01180 ;
               01190 ; CONVERT TAB TO ONE
               01200 ;
               01210 ;
```

Program continues

```
94AE FE09     01220 NOTAB   CP      9               ;Tab
94B0 2002     01230         JR      NZ,JJ
94B2 3E01     01240         LD      A,1
94B4 F5       01250 JJ      PUSH    AF              ;Instructions replaced by JP
94B5 E5       01260         PUSH    HL
94B6 21BA41   01270         LD      HL,41BAH
94B9 C33E47   01280         JP      473EH
              01290 ;
              01300 ;
              01310 ; INTERCEPT 'T' OUTPUT, MODIFY, AND SEND TO PRINTER
              01320 ;
              01330 ;
94BC 79       01340 CHANGE  LD      A,C
94BD FE0D     01350         CP      13              ;Ignore line feeds
94BF C8       01360         RET     Z
94C0 FE23     01370         CP      '#'             ;Start of stop title
94C2 CAD195   01380         JP      Z,TITLE
94C5 3A9293   01390         LD      A,(TYPEA+3)     ;Character in title?
94C8 B7       01400         OR      A
94C9 C2D195   01410         JP      NZ,TITLE
94CC 79       01420         LD      A,C
94CD FE01     01430         CP      1               ;Tab
94CF CA9295   01440         JP      Z,PARA
              01450 ;
              01460 ;
              01470 ; CAPITOLIZATION RDUTINE
              01480 ;
              01490 ;
              01500 ; Replace the next command with NOP, NOP, NOP if you did not
              01510 ; add the lower case modification earlier.
94D2 C3D495   01520         JP      EASY            ;No capitolization
94D5 3A9193   01530         LD      A,(TYPEA+2)     ;Start of sentence?
94D8 FE20     01540         CP      ' '
94DA 204D     01550         JR      NZ,ONWARD
94DC 3A9093   01560         LD      A,(TYPEA+1)
94DF FE20     01570         CP      ' '
94E1 2046     01580         JR      NZ,ONWARD
94E3 3A8F93   01590         LD      A,(TYPEA)
```

```
94E6 FE2E    01600          CP      '.'
94E8 2804    01610          JR      Z,YES
94EA FE3F    01620          CP      '?'
94EC 203B    01630          JR      NZ,ONWARD
94EE CD1E95  01640 YES      CALL    CAP
94F1 1803    01650          JR      REG1
94F3 CD1295  01660 REGULA   CALL    REVERS          ;Routine for regular letter
94F6 CD0595  01670 REG1     CALL    MOVE
94F9 79      01680          LD      A,C
94FA FE49    01690          CP      'I'
94FC 2001    01700          JR      NZ,XX
94FE 3C      01710          INC     A
94FF 329193  01720 XX       LD      (TYPEA+2),A
9502 C3D495  01730          JP      EASY
9505 3A9093  01740 MOVE     LD      A,(TYPEA+1)
9508 328F93  01750          LD      (TYPEA),A
950B 3A9193  01760          LD      A,(TYPEA+2)
950E 329093  01770          LD      (TYPEA+1),A
9511 C9      01780          RET
9512 79      01790 REVERS   LD      A,C             ;Correct keyboard
9513 FE41    01800          CP      41H
9515 D8      01810          RET     C
9516 FE5B    01820          CP      5BH
9518 3004    01830          JR      NC,CAP
951A C620    01840          ADD     A,20H
951C 4F      01850          LD      C,A
951D C9      01860          RET
951E 79      01870 CAP      LD      A,C
951F FE61    01880          CP      61H             ;Capitolize
9521 D8      01890          RET     C
9522 FE7B    01900          CP      7BH
9524 D0      01910          RET     NC
9525 D620    01920          SUB     20H
9527 4F      01930          LD      C,A
9528 C9      01940          RET
9529 3A9193  01950 ONWARD   LD      A,(TYPEA+2)
952C FE49    01960          CP      'I'
```

Program continues

↙ *Reader Service—see page 274*

```
952E 2B07      01970          JR      Z,EIP           ;Previous letter 'i'
9530 79        01980          LD      A,C
9531 FE49      01990          CP      'I'
9533 2819      02000          JR      Z,EI            ;Current letter 'i'
9535 18BC      02010          JR      REGULA
9537 79        02020 EIP      LD      A,C
9538 FE20      02030          CP      ' '
953A 2009      02040          JR      NZ,GG
953C C5        02050          PUSH    BC
953D 0E49      02060          LD      C,'I'
953F CDD495    02070          CALL    EASY
9542 C1        02080          POP     BC
9543 18AE      02090          JR      REGULA
9545 C5        02100 GG       PUSH    BC
9546 0E69      02110          LD      C,'I'+20H
9548 CDD495    02120          CALL    EASY
954B C1        02130          POP     BC
954C 18A5      02140          JR      REGULA
954E 3A9193    02150 EI       LD      A,(TYPEA+2)
9551 FE20      02160          CP      ' '
9553 209E      02170          JR      NZ,REGULA
9555 3E49      02180          LD      A,'I'
9557 329193    02190          LD      (TYPEA+2),A
955A C9        02200          RET
               02210 ;
               02220 ;
               02230 ; LINEFEED ROUTINE
               02240 ;
               02250 ;
955B CD6695    02260 LINEFD   CALL    LINEF1
955E 3A6595    02270          LD      A,(DOUBLE)
9561 B7        02280          OR      A
9562 C8        02290          RET     Z
9563 1801      02300          JR      LINEF1
               02310 ; Insert zero in the next address for single space and one for
               02320 ; double space.
9565 00        02330 DOUBLE   DEFB    0
9566 0E0D      02340 LINEF1   LD      C,13            ;Print line feed
9568 CD8692    02350          CALL    PRINT
956B C9        02360          RET
               02370 ;
               02380 ;
               02390 ; ROUTINE TO PRINT HALF-FILLED BUFFER
               02400 ;
               02410 ;
956C 3A9393    02420 STUFF    LD      A,(TYPEA+4)
956F B7        02430          OR      A
9570 C8        02440          RET     Z
9571 E5        02450          PUSH    HL
9572 C5        02460          PUSH    BC
9573 CDBD95    02470          CALL    MAR1
9576 47        02480          LD      B,A
9577 219A93    02490          LD      HL,BUFF
957A 4E        02500 YA       LD      C,(HL)
957B CDB692    02510          CALL    PRINT
957E 23        02520          INC     HL
957F 10F9      02530          DJNZ    YA
9581 3E00      02540          LD      A,0
9583 329393    02550          LD      (TYPEA+4),A
9586 219A93    02560          LD      HL,BUFF
9589 229493    02570          LD      (TYPEA+5),HL
958C CD5B95    02580          CALL    LINEFD
958F C1        02590          POP     BC
9590 E1        02600          POP     HL
9591 C9        02610          RET
               02620 ;
               02630 ;
               02640 ; PARAGRAPH ROUTINE
               02650 ;
               02660 ;
9592 CD6C95    02670 PARA     CALL    STUFF
9595 3ABB95    02680          LD      A,(EXTRA)
9598 B7        02690          OR      A
9599 2805      02700          JR      Z,NOEX
959B 0E0D      02710          LD      C,13
959D CD8692    02720          CALL    PRINT
95A0 E5        02730 NOEX     PUSH    HL
95A1 C5        02740          PUSH    BC
95A2 3ABC95    02750          LD      A,(PARAL)
95A5 47        02760          LD      B,A
95A6 219A93    02770          LD      HL,BUFF
95A9 0E20      02780          LD      C,' '
95AB 71        02790 RA       LD      (HL),C
```

Program continues

```
95AC 23       02800          INC     HL
95AD 10FC     02810          DJNZ    RA
95AF 229493   02820          LD      (TYPEA+5),HL
95B2 3ABC95   02830          LD      A,(PARAL)
95B5 329393   02840          LD      (TYPEA+4),A
95B8 C1       02850          POP     BC
95B9 E1       02860          POP     HL
95BA C9       02870          RET
              02880 ; Insert zero in the next address for no extra lines between
              02890 ; paragraphs, and one for one extra line.
95BB 00       02900 EXTRA    DEFB    0
              02910 ; Insert below the number of spaces in a paragraph indentation.
95BC 05       02920 PARAL    DEFB    5
              02930 ;
              02940 ;
              02950 ; MARGIN ROUTINE
              02960 ;
              02970 ;
95BD F5       02980 MAR1     PUSH    AF
95BE C5       02990          PUSH    BC
95BF 3AD095   03000          LD      A,(MARGIN)
95C2 B7       03010          OR      A
95C3 2808     03020          JR      Z,KLK
95C5 47       03030          LD      B,A
95C6 0E20     03040          LD      C,20H
95C8 CD8692   03050 XJ       CALL    PRINT
95CB 10FB     03060          DJNZ    XJ
95CD C1       03070 KLK      POP     BC
95CE F1       03080          POP     AF
95CF C9       03090          RET
              03100 ; Load the next address with the number of spaces in the left
              03110 ; margin.  This should be zero for the Quick Printer II and
              03120 ; about 12 for a regular printer.
95D0 0C       03130 MARGIN   DEFB    12
95D1          03140 TITLE    EQU     $
95D4          03150 EASY     EQU     TITLE+3
0000          03160          END
00000 TOTAL ERRORS
```

style, etc. However, comments which begin in the comment column can be ignored; they explain the operation of the code for those who are interested.

Memory Distribution

See Table 2 for an explanation of memory distribution. When EDTASM is entered, it occupies 7000-96E6. Immediately, 7000-89FF is moved to the original EDTASM location; some of this area then becomes part of the editor buffer. The BASIC RAM area 4000-42FF is modified by EDTASM, but temporarily restored to original form by the J command.

A Final Touch

The modified EDTASM can be used to write and correct assembly language programs interactively, much as BASIC is used. There is one difference. When assembly language programs go wrong, they go horribly wrong. If your program has a mistake, the J command may not return control to the assembler and you might have to push RESET to tame the computer.

In that case, there is a simple way to return to the editor/assembler with your program intact. After you hit RESET, the DOS prompt will appear. Execute SAVE and EDTASM and Y. You will find yourself back in the assembler with the editor buffer intact.

The program SAVE/CMD used above should be permanently placed on the EDTASM disk. Create it by using DEBUG and entering the following machine code.

```
8000  F3 21 F9 5C 11 C0 9F 01
8008  07 23 ED B0 C3 2D 40
```

Owners of version 1.2 should replace F9 on the top line with F0 and 07 on the bottom line with 10.

Turn off DEBUG, and execute:

```
DUMP SAVE/CMD (START = X'8000',END =
'8010',TRA = X'8000')■
```

9327	Zero for Parallel Printer, One for Serial Printer
9329	Current Line, Zero for Top of Form
932A	Lines per Page
932C	Zero for Continuous Printing, One for Pause between Pages
9399	Number of Printed Characters per Line
9565	Zero for Single Space, One for Double Space
95BB	Zero for No Additional Spaces between Paragraphs, One for Additional Spaces
95BC	Length of Paragraph Indentations
95D0	Length of Left Margin
96E5	Zero for Single Width Letters in Titles, One for Double Width Letters

Table 1.

0000-3FFF	Basic ROM, Keyboard, Etc.
4000-42FF	Basic Fixed RAM, Used Independently by EDTASM
4300-5CF8	EDTASM (Original Program)
5CF9-7FFF	Text Buffer 5C F9
8000-89FF	Used Only When Loading EDTASM
8A00-96E6	EDTASM (New Program)
96E7-98FF	Available for Future Expansion
9900-99EF	Stack While Using J
99F0-9FFF	Used by J, X, Y
A000-BFFF	Reserved for Machine Language Programs, Used by X, Y
C000-FFFF	Unused

Table 2. Memory Distribution

Program Listing 8. EDTASM 8

```
                00100 ; INSTRUCTIONS:  Run EDTASM7.  Enter the following text and
                00110 ; assemble it to memory.  Then read the instructions at the
                00120 ; end of this section.
                00130 ;
                00140 ;
                00150 ; PRINT ROUTINE
                00160 ;
                00170 ;
95D1            00180 TITLE   EQU     95D1H
95D4            00190 EASY    EQU     95D4H
955B            00200 LINEFD  EQU     955BH
9399            00210 BUFFL   EQU     9399H
956C            00220 STUFF   EQU     956CH
9286            00230 PRINT   EQU     9286H
938F            00240 TYPEA   EQU     938FH
939A            00250 BUFF    EQU     939AH
941A            00260 BUFFT   EQU     941AH
95BD            00270 MAR1    EQU     95BDH
95D1            00280         ORG     TITLE
95D1 C33F96     00290         JP      TITLE1
95D4            00300         ORG     EASY
95D4 79         00310         LD      A,C             ;Don't print initial space
95D5 FE20       00320         CP      ' '
95D7 2005       00330         JR      NZ,XA
95D9 3A9393     00340         LD      A,(TYPEA+4)
95DC B7         00350         OR      A
95DD C8         00360         RET     Z
95DE E5         00370 XA      PUSH    HL
95DF D5         00380         PUSH    DE
95E0 C5         00390         PUSH    BC
95E1 79         00400         LD      A,C             ;Insert character into buffer
95E2 2A9493     00410         LD      HL,(TYPEA+5)
95E5 77         00420         LD      (RL),A
95E6 23         00430         INC     HL
95E7 229493     00440         LD      (TYPEA+5),HL
95EA 3A9393     00450         LD      A,(TYPEA+4)
95ED 3C         00460         INC     A
95EE 329393     00470         LD      (TYPEA+4),A
95F1 219993     00480         LD      HL,BUFFL
95F4 BE         00490         CP      (HL)            ;Is buffer full?
```

Program continues

```
95F5 2804      00500          JR      Z,XB
95F7 C1        00510 STOP     POP     BC
95F8 D1        00520          POP     DE
95F9 E1        00530          POP     HL
95FA C9        00540          RET
95FB CDBD95    00550 XB       CALL    MAR1
95FE 46        00560          LD      B,(HL)          ;Find last space in buffer
95FF 2A9493    00570          LD      HL,(TYPEA+5)
9602 2B        00580 XC       DEC     HL
9603 7E        00590          LD      A,(HL)
9604 FE20      00600          CP      ' '
9606 2806      00610          JR      Z,XD
9608 10F8      00620          DJNZ    XC
960A 219993    00630          LD      HL,BUFFL
960D 46        00640          LD      B,(HL)          ;If no spaces, print entire line
960E C5        00650 XD       PUSH    BC
960F 219A93    00660          LD      HL,BUFF         ;Print buffer
9612 4E        00670 XD1      LD      C,(HL)
9613 CD8692    00680          CALL    PRINT
9616 23        00690          INC     HL
9617 10F9      00700          DJNZ    XD1
9619 C1        00710          POP     BC
961A 3A9993    00720          LD      A,(BUFFL)
961D 90        00730          SUB     B               ;Remaining buffer length
961E B7        00740          OR      A
961F 329393    00750          LD      (TYPEA+4),A
9622 2810      00760          JR      Z,XE            ;Remaining buffer empty
9624 47        00770          LD      B,A             ;Move old buffer to new
9625 119A93    00780          LD      DE,BUFF
9628 7E        00790 XF       LD      A,(HL)
9629 12        00800          LD      (DE),A
962A 23        00810          INC     HL
962B 13        00820          INC     DE
962C 10FA      00830          DJNZ    XF
962E ED539493  00840          LD      (TYPEA+5),DE
9632 1806      00850          JR      XG
9634 219A93    00860 XE       LD      HL,BUFF
9637 229493    00870          LD      (TYPEA+5),HL
963A CD5B95    00880 XG       CALL    LINEFD
963D 18B8      00890          JR      STOP
               00900 ;
               00910 ;
               00920 ; TITLE ROUTINE
               00930 ;
               00940 ;
963F 3A9293    00950 TITLE1   LD      A,(TYPEA+3)     ;Start of title?
9642 FE00      00960          CP      0
9644 2013      00970          JR      NZ,AA
9646 E5        00980          PUSH    HL
9647 3E01      00990          LD      A,1
9649 329293    01000          LD      (TYPEA+3),A     ;Title mode
964C 3E00      01010          LD      A,0
964E 329693    01020          LD      (TYPEA+7),A     ;Title empty
9651 211A94    01030          LD      HL,BUFFT
9654 229793    01040          LD      (TYPEA+8),HL    ;Start of title
9657 E1        01050          POP     HL
9658 C9        01060          RET
9659 79        01070 AA       LD      A,C             ;End of title?
965A FE23      01080          CP      '#'
965C 282C      01090          JR      Z,AB
965E E5        01100          PUSH    HL
965F 3A9693    01110          LD      A,(TYPEA+7)     ;If not, add letter to title
9662 47        01120          LD      B,A
9663 3AE596    01130          LD      A,(DW)
9666 B7        01140          OR      A
9667 281B      01150          JR      Z,AP
9669 219993    01160          LD      HL,BUFFL
966C 7E        01170          LD      A,(HL)
966D CB3F      01180          SRL     A
966F B8        01190 AQ       CP      B
9670 2810      01200          JR      Z,HH            ;unless title full
9672 2A9793    01210          LD      HL,(TYPEA+8)
9675 79        01220          LD      A,C
9676 77        01230          LD      (HL),A
9677 23        01240          INC     HL
9678 229793    01250          LD      (TYPEA+8),HL
967B 3A9693    01260          LD      A,(TYPEA+7)
967E 3C        01270          INC     A
967F 329693    01280          LD      (TYPEA+7),A
9682 E1        01290 HH       POP     HL
9683 C9        01300          RET
9684 219993    01310 AP       LD      HL,BUFFL
9687 7E        01320          LD      A,(HL)
```

Program continues

```
9688 18E5     01330          JR      AQ
968A AF       01340 AB       XOR     A                ;Leave title mode
968B 329293   01350          LD      (TYPEA+3),A
968E CD6C95   01360          CALL    STUFF            ;Print all before title
9691 3A9693   01370          LD      A,(TYPEA+7)
9694 B7       01380          OR      A                ;Title empty?
9695 2004     01390          JR      NZ,AC
9697 CD5B95   01400          CALL    LINEFD           ;If so, print empty line
969A C9       01410          RET
969B F5       01420 AC       PUSH    AF
969C CDBD95   01430          CALL    MAR1
969F 3AE596   01440          LD      A,(DW)
96A2 B7       01450          OR      A
96A3 2805     01460          JR      Z,AD
              01470 ; If you can print double width characters, load C below with
              01480 ; the control character which begins double width printing.
              01490 ; This control character is 0FH for the Radio Shack Quick
              01500 ; Printer II.
96A5 0E0F     01510          LD      C,0FH
96A7 CD8692   01520          CALL    PRINT
96AA F1       01530 AD       POP     AF
96AB 47       01540          LD      B,A
96AC 3A9993   01550          LD      A,(BUFFL)
96AF 4F       01560          LD      C,A
96B0 3AE596   01570          LD      A,(DW)
96B3 B7       01580          OR      A
96B4 2805     01590          JR      Z,AM
96B6 79       01600          LD      A,C
96B7 CB3F     01610          SRL     A
96B9 1801     01620          JR      AN
96BB 79       01630 AM       LD      A,C
96BC 90       01640 AN       SUB     B                ;Spaces at two ends of title
96BD CB3F     01650          SRL     A                ;Divide by two
96BF 47       01660          LD      B,A
96C0 B7       01670          OR      A
96C1 2807     01680          JR      Z,AE             ;No spaces at left
96C3 0E20     01690          LD      C,' '
96C5 CD8692   01700 AJ       CALL    PRINT
96C8 10FB     01710          DJNZ    AJ
96CA 3A9693   01720 AE       LD      A,(TYPEA+7)      ;Title length
96CD 47       01730          LD      B,A
96CE E5       01740          PUSH    HL
96CF 211A94   01750          LD      HL,BUFFT
96D2 4E       01760 AG       LD      C,(HL)           ;Print title
96D3 CD8692   01770          CALL    PRINT
96D6 23       01780          INC     HL
96D7 10F9     01790          DJNZ    AG
96D9 E1       01800          POP     HL
96DA CD5B95   01810          CALL    LINEFD
              01820 ; Replace the next line with NOP, NOP if your printer is not
              01830 ; automatically converted from double width to single width
              01840 ; mode by a line feed.  Notice that the line should remain as
              01850 ; is for the Radio Shack Quick Printer II.
96DD 1805     01860          JR      AH
              01870 ; Load C below with the control character needed to convert
              01880 ; your printer back to single width style.
96DF 0E00     01890          LD      C,00
96E1 CD8692   01900          CALL    PRINT
96E4 C9       01910 AH       RET
              01920 ; Insert a zero in the address below for single width title
              01930 ; characters and a one for double width title characters.
96E5 00       01940 DW       DEFB    0
96E6          01950 CONT5    EQU     $
              01960 ;
              01970 ;
              01980 ; INSTRUCTIONS:  Now a few cleanup details before saving the
              01990 ; final version of EDTASM.  Make sure locations INIT, LINE, PAGE,
              02000 ; TOTAL, and PAUSE contain the right values (they will unless
              02010 ; you used the printer after running EDTASM7 or you want different
              02020 ; initial values than those I provided).  Next POKE 8AB1,00
              02030 ; and POKE 8AB2,00 to protect the assembler from assembling
              02040 ; code on top of itself.  Then POKE 7397,7F to enlarge the
              02050 ; text buffer and POKE 8E4D,23 to modify SIZE accordingly.
              02060 ; Make sure locations BUFFL, DOUBLE, EXTRA, PARAL, MARGIN,
              02070 ; and DW contain the values you want on initialization.  Issue
              02080 ; the command M1 if you want EDTASM to power up in keyboard
              02090 ; entry mode one (and M0 or M2 for other entry modes).  Finally,
              02100 ; if you added the lower case modification, put the assembler
              02110 ; in the mode in which you wish it to power up.  Issue the command
              02120 ; DUMP EDTASM8/CMD (START=X'7000',END=X'96E6',TRA=X'8A00').
0000          02130          END
00000 TOTAL ERRORS
```

A little drawing program.

Compu-Sketch

Merl J. Hendricks
140 Gradolph St.
Toledo, OH 43612

The February 1980 issue of *80 Microcomputing* carried an article by Wes Thielke on "ROM Routines." The section on keyboard encoding interested me. I entered the program and ran it.

After looking at the output, I thought I could use it to create a little drawing program. I came up with the following program.

Compu-sketch

When an arrow key is depressed, the pixel will move in the indicated direction. If two keys are pressed at the same time, the pixel will move in the diagonal direction determined by the keys.

Interesting, but I thought I should be able to blank the trail so I could interrupt the lines. After some playing around, I set it up so that if the space bar is held down while the arrow key is depressed, the pixel will not leave a trail.

Lastly, I could not clear the screen unless I used the break key, and then ran the program again. With a couple of added lines I cleared the screen by pressing the clear key and space bar at the same time.

Graphic Set-ups

This little program could certainly speed up game graphic set-ups.

A very short PEEK-POKE routine could transfer the graphics from the screen to memory and another PEEK-POKE routine could recall them. Take a look at the program. ■

```
100  CLS:X = 56 Y = 22.DEFINT C
110  C = PEEK(14400)
120  IF C AND 8 THEN Y = Y - 1
130  IF C AND 16 THEN Y = Y + 1
140  IF C AND 32 THEN X = X - 1
150  IF C AND 64 THEN X = X + 1
160  IF X> 127 THEN X = X - 1
170  IF X < 1 THEN X = X + 1
180  IF Y> 47 THEN Y = Y - 1
190  IF Y< 0 THEN Y = Y + 1
200  IF C = 130 THEN PRINT CHR$(2) GOTO110
210  IF C> 120 THEN RESET(X,Y).GOTO110
220  RESET (X,Y) FOR T = 1TO1.NEXT.SET (X Y) GOTO110
230  END
```

Program Listing 1. Etch-a-compu-sketch.

Tell your data printer where to go, with software, of course.

RESTORE
Data Pointer Control

David R. Cecil
Dept. of Mathematics
Texas A & I University
Kingsville, TX 78363

```
 10 ON ERROR GOTO 200
 20 D=0: EL=0
 30 READ A$: D=D+1
 40 PRINT A$;
 50 IF A$="MOUSE" THEN EL=D ELSE 30
 55 REM "MOUSE" IS THE LAST OF THE STRING DATA
 60 READ A: PRINT A,
 70 GOTO 60
100 DATA CAT, DOG, MOUSE
110 DATA 9, 18, -21, 3.98
200 RESTORE
210 FOR I=1 TO EL
220 READ A$: NEXT I
230 RESUME 50
```

Program Listing 1

```
 10 ON ERROR GOTO 200
 20 D=0: EL=0: REM D FOR DELETE, EL FOR LAST ELIMINATED
 30 READ A: D=D+1
 40 IF A=9.9 THEN EL=D: REM 9.9 IS LAST DATA ELIMINATED
 50 PRINT A,
 60 GOTO 30
100 DATA 6, 9, -8, 9.9, 4, 8
110 DATA 7, 3.45, 2.1
200 RESTORE          : REM ALL DATA IS RESTORED
210 FOR I=1 TO EL    : REM THE FIRST EL DATA ARE READ
220 READ A: NEXT I   : REM BUT NOT PRINTED
230 RESUME
```

Program Listing 2

```
65000 INPUT M, N, P, R              : REM SEE TEXT FOR M, N, P, R VALUES
65010 RR=N+(R-1)*(N-P-M)            : REM RR=TOTAL DATA WANTED
65020 DIM A(RR)                     : REM THIS SIZE ARRAY ALLOWS R
                                      REPEATS
65030 FOR I=1 TO N                  : REM THE FIRST N DATA ITEMS ARE
                                      READ
65040 READ A(I)                     : REM AND PLACED IN THE FIRST N
65050 NEXT I                        : REM ENTRIES OF AN ARRAY A
65060 FOR I=1 TO RR                 : REM ALL OF ARRAY A IS FORMED
65070 IF I>M THEN A(I)=A(I-(N-M)*INT((I-M-1)/(N-M)))
65080 REM THIS INSURES THE FIRST M ITEMS ARE USED ONLY ONCE
65090 IF I>2*N-M-P THEN A(I)=A(I+M+P-N)
65100 REM THIS ENABLES US TO USE THE LAST P ITEMS ONLY ONCE
65110 NEXT I
65120 RETURN
```

Program Listing 3

Methods to effectively set the DATA pointer to some other place than the start of the DATA list or only to certain data statements through the RESTORE command have been the topic of many a letter to the editor in microcomputing journals of late.

The following programs illustrate two methods of obtaining a selected RESTORE. The method in Programs 1 and 2 RESTORES to the first of the data set and then reads to the desired data. The method in Program 3 places the data in an array and relabels the subscripts of the array, so that selected lines of data may be restored and used as often as desired.

When we run Program 1, the output will be 6 9 -8 9.9 4 8 7 3.45 2.1 4 8 7 3.45 2.1 4, etc. We have restored only the data appearing after the data value of 9.9.

Program Listing 2 allows us to restore only numeric data, if we have all the string data listed together and appearing before any numeric data.

If all the numeric data is listed before any string data, then we simply replace each A$ by A and each A by A$, as in Program 2. Line 50 would have A = last numeric data value.

Using an Array

The second method, using an array for storing the data, allows us to use all the data once and then reuse any consecutive portion as often as we wish. To illustrate this method, we'll use a subroutine with high line numbers.

To use this Data Restore Computation method with any of your programs, type the subroutine in Program Listing 3 and, for the very first line of your existing program, put 0 GOSUB 65000.

The subroutine asks for four inputs M, N, P and R. These values are given by the following:

M = the number of data items at the beginning of the DATA statements that will be used only once;

N = the number of data items to be used until we want to reuse some data;

P = the number of data items, from item N − P + 1 to N, that will be used only once (the middle items from the M + 1st through the N − Pth are the only items that will be reused);

R = repeats the total number of times the data items M + 1 through N − P will be used.

Table 1 illustrates how the subroutine forms array A. The

Original Subscripts	Relabeled Subscripts	Comments
A(1)	A(1)	This data only used at first and not repeated.
⋮	⋮	
A(M)	A(M)	
A(M + 1)	A(M + 1)	First use of that part of the data to be repeated
⋮	⋮	
A(N – P)	A(N – P)	
A(N – P + 1)	A(N – P + 1)	This data used only once and not repeated
⋮	⋮	
A(N)	A(N)	
A(N + 1)	A(M + 1)	
⋮	⋮	
A(2N – M – P)	A(N – P)	R – 1 additional uses of data items M + 1 thru N – P
⋮	⋮	
A(RR – N + P + M + 1)	A(M + 1)	
⋮	⋮	
A(RR)	A(N – P)	

Table 1

short program in Listing 4 with four DATA statements should help you get used to our M, N, P, R input notation.

If we input 4, 10, 2, 4 when the prompt ? appears, then data line 20 will be used only once (the 4 input); all the data will be used (10 items); data line 50 will be used only once (the 2 input); and data lines 30 and 40 will be used a total of four times (the 4 input). The output will then be (4 9 -8 0) 7 1 5 2 (3 6) (7 1 5 2) (7 1 5 2) (7 1 5 2). The parentheses are not output, but are used to show the grouping.

If we want to use data lines 20 and 30, with line 30 used six times, we would input 4, 7, 0, 6. (Note that line 30 is used once for the first listing of seven items and then line 30 is used five additional times.)

We see that 0, 10, 0, 3 would use all ten data items three times, while 4, 8, 1, 2 would use data lines 20, 30 and 40, and then repeat data line 30 one more time.

Try other combinations of values for inputs. Be sure you use whole number values for all four inputs with M ≥ 0, N > 0, P ≥ 0, R > 0 and N > M + P. Program Listing 4 has very little data, but remember that for larger N and R, the subroutine takes a longer time to form the array.

Program Listing 3 can be used equally well with string data. You need only change each A to A$ in lines 65020, 65040, 65070 and 65090. You could try this with Program Listing 5.

Inputs of 3, 8, 2, 4 give an output of ABCDEFGHDEFDEFDEF. If we change the first part of line 10 to FOR I = M + 1 to RR, then inputs of 2, 5, 0, 3 would result in line 30, and only line 30, being used three times.

I hope these methods will make your RESTORE problems seem like child's play. ■

```
 0 GOSUB 65000
10 FOR I = 1 TO RR: PRINT A(I);: NEXT I: STOP
20 DATA 4, 9, -8, 0
30 DATA 7, 1, 5
40 DATA 2
50 DATA 3,6
```

Program Listing 4

```
 0 GOSUB 65000
10 FOR I = 1 TO RR: PRINT A$(I);: NEXT I: STOP
20 DATA A, B
30 DATA C, D, E
40 DATA F
50 DATA G, H, I, J
```

Program Listing 5

An LPUNK list program for Model IIs.

Less Is More

C. E. Winterbauer
3910 Bandini St.
San Diego, CA 92103

Sometimes a program is written and becomes a favorite for a silly reason.

This is partly the case with this routine. While it's useful, it's particularly attractive because the program does so much with so little code! It was written in assembly language but is so short that it's faster to load it directly using Debug. It also uses a relative jump, making its location unimportant. I placed it at 3000H only for convenience while using it alone.

It's an easy program.

```
3000                    ORG     3000H
3000 2100F8     PRINTER LD      HL,0F800H       ;MEM START LOCATION
3003 0618               LD      B,24            ;INITIAL NUM OF LINES
3005 C5         PRTLNE  PUSH    BC              ;SAVE IT FOR LATER
3006 3E81               LD      A,81H           ;VALUE FOR DSPLY TURN ON
3008 D3FF               OUT     (0FFH),A        ;DSPLY MEM SWITCH PORT
300A 0650               LD      B,80            ;NUM CHARS/LINE
300C 0E0D               LD      C,0DH           ;CARR RET ADVANCES PRINTER
300E 3E13               LD      A,19            ;SUPERVISOR CALL CODE
3010 CF                 RST     8               ;EXECUTES SUPERVISOR
3011 115000             LD      DE,80           ;INCREMENT TO NEXT LINE
3014 19                 ADD     HL,DE           ;SET UP NEW TRANSFER LOC
3015 C1                 POP     BC              ;GET CURRENT LINE NUM
3016 10ED               DJNZ    PRTLNE          ;CHECK,JMP BACK IF NOT DONE
301A D3FF               OUT     (0FFH),A        ;DSPLY MEM SWITCH PORT
301C 3E24               LD      A,36            ;SUPERVISOR CODE JMP TRSDOS
301E CF                 RST     8               ;EXECUTES SUPERVISOR
0000                    END
```

Model II Machine Code for Printing the Display Contents

The number of lines to be printed can be easily changed (3004H). So can the end of the program, both by adding a form feed instruction, or returning or jumping to another routine instead of back to TRSDOS. I use this routine in a larger program, which examines and edits specified sectors of the disk. I call this routine whenever I want a copy of the information (sector data) on the screen. Of course, my printer has already been initialized and is on line.

One of the key points of the routine is the knowledge of the port and the value sent to that port to perform the operation. I have a 64K Model II and the turn-on value is 81H and the turn-off value is 0.

I hope this routine will be useful for those needing a simple but effective print routine when coding in assembly or machine language for the Model II. ■

Computer dating with a difference.

Gregorian Converter

Hubert C. Borrmann
2840 S. Circle Dr.
Colorado Springs, CO 80906

When we work with dates in real life, we look at our digital watch or calendar. Whenever we have to compare several dates, we let our fingers do a lot of walking through the calendar, because we cannot use simple arithmetic when subtracting one date from another.

We use the Gregorian Calendar which was established in the early Middle Ages. Our computer likes its dates simpler, however. Some large data processing installations are using the Julian date, named after good old Julius Caesar. The Julian date consists of five digits, the first two of which are the year, followed by a three digit day within that year. For example: 80039 is Feb. 8, 1980; 80061 is March 1, 1980.

When comparing two Julian dates within the same year, we merely subtract the lower from the higher and have the elapsed days. If we go from one year to the next we first determine how many days are left in the current year, and then how far we want to go into the next year, and add the values together.

This program will convert a Gregorian date to a Julian date, or vice versa, via two entry points. To convert to a Julian date we furnish the subroutine with three values: the month from 1-12 in 'MM', the day from 1-31 in 'DD', and the year from 10-99 in 'YY'.

GOSUB 12100 and the subroutine will return the five-digit Julian date in the form YYDDD in variable 'JD'. To convert to a Gregorian date furnish the subroutine with the Julian date in 'JD' from 10001-99365, then GOSUB 12200, and the subroutine returns with the Gregorian month in 'MM', the Gregorian day in 'DD' and the year in 'YY'.

Both entry points edit the entered data, and if they are in error, a message is printed and the return variables contain zero.

Look at the subroutine in Program Listing 1. There is a rather extensive REM section from lines 12000 to 12090. If you are pressed for space this part may be eliminated. In addition to the variables 'MM', 'DD', 'YY' and 'JD', which are not destroyed, I am using the Y-family in this subroutine. 12100-12180 is the Gregorian-to-Julian routine, and we first link to another subroutine at 12300, which loads our table into variable 'VV$', a subroutine within a subroutine.

Line 12110 edits the furnished Gregorian month, day and year within valid ranges. Any errors cause the logic to go to line 12180 which prints ERROR, sets variable 'JD' to zero and goes to line 12170, which is a RETURN statement.

The table in 'VV$', (line 12300), consists of 13 elements, one for each month of the year, and one terminetor. Each element is five positions long, its first two positions indicating how many days each month has. The remaining three positions describe how far into the year the first day of the month is. The first element, 31001, is January. It tells us that January has 31 days and the first of January is the first day of the year (001). The next element, 28032, tells us that February has 28 days and February 1st is the 32nd day of the year. March 1st is day 60 of the year, etc. The table is adjusted for Leap years as well. A leap year is determined by dividing the year by four, and it there is no remainder, it is a leap year (see lines 12120, 12160, 12235 and 12240). If the test in line 12120 is not true, line 12130 is executed, and here we check that the submitted day is not larger than the last day of the month. We find the correct table element by multiplying the month by five and then subtracting four. This points to position one of the element, which is also position one of the number of days in this month.

Let's say, for example, the submitted month was 04 and the day was 31. We multiply four times five and subtract four = 16. The statement in line 12130 would say: If the day entered (31) is larger than the two positions in the string (the table) VV$, starting at position 16, then it is an error. Counting off the 16th and 17th position, you find 30, indicating that the 31st of April is an error.

We have to use VAL because the day variable 'DD' is numeric and strings are alphanumeric. If the check passes, in line 12140 the Julian day of the first of the month is picked up. If April 30th was submitted, again we find our month element and we are

pointing at position one of the three position day within the year field; 091 for April, to which we add the day value 30, and since we do not want to count the first day twice, we subtract one. The Julian day for the 30th of April is 120.

Line 12150 develops the five position Julian date by multiplying the submitted year by 1000 and adding the Julian day to it. It is in variable 'JD' and we can then go back to the main routine. Before we RETURN we test for a leap-year and adjust our answer accordingly.

For any printouts and communication with the "other world" we need to reconvert Julian dates to Gregorian dates. Line 12200 is the entry point for the Julian to Gregorian conversion. Line 12210 resets some variables and also converts the submitted numeric Julian day in 'JD' to a string. If the Julian date is five positions long (as it should be) we will have six positions in our string V7$.

The first position is for the sign. Line 12220 picks the five significant positions out of V7$ and builds V6$. We also split V6$ into the year in 'YY' and the day in 'DD'. In line 12240 variable V4 is set up with a one if we are working with a leap year and the submitted day is larger than 59. Otherwise the value in V4 remains zero.

Lines 12250 and 12260 form a loop, checking each table element's Julian day until a month is found whose entry is larger than the submitted day in 'DD'. On checking each entry the value in V4 is added to it and this has the same effect as if all day fields from March on are larger by one in the case of leap year. On leaving the loop we are one month too far, and in line 12270 in V3 the day element of the preceding month is picked up. If this picked up day is larger than 32, it's added to the value in V4, which in the case of a leap year has the same effect as if the day fields from March-December were larger by one.

In line 12280 we subtract the picked up day in V3 plus one from the submitted Julian day, and the result is the desired Gregorian day in 'DD'. We know that we went one month too far and reduce the month counter V5 by one, which goes into 'MM'. The year has been in 'YY' since line 12220. Line 12290 returns to the main program, and line 12295 is the error line.

To try the subroutine, see program listing 2 for a small driver which alternates between the two entry points, and also checks for a zero return. ■

```
12000 'THIS DATE-CONVERSION SUBROUTINE HAS 2 ENTRY POIN
      TS :
12005 '   1)    12100 CONVERTS FROM GREGORIAN TO JULIAN
      DATE
12010 '   2)    12200 CONVERTS FROM JULIAN TO GREGORIAN
      DATE
12015 '
12020 'WHEN CONVERTING TO JULIAN DATE, SUPPLY THE GREGO
      RIAN
12025 'DATE AS FOLLOWS : MONTH  (1-12) IN VARIABLE 'MM'
12030 '                  DAY    (1-31)             'DD'
12035 '                  YEAR   (00-99)            'YY'
12040 'AND THE JULIAN DATE WILL BE IN VARIABLE 'JD' AT
      EXIT
12045 'AS YYDDD (YY=YEAR, DDD=DAY WITHIN YEAR)
12050 '   EXAMPLE : 02,00,00  (FEB.0, 1900) IS 00030
12055 '             03,01,00  (MAR.1, 1900) IS 00061
      ETC.
12060 '
12065 'WHEN CONVERTING TO GREGORIAN DATE, SUPPLY THE JU
      LIAN
12070 'DATE IN VARIABLE 'JD' AT ENTRY, AND AT EXIT THE
      VARIABLES
12075 '                  'MM' WILL CONTAIN THE GREGORIAN
       MONTH
12080 '                  'DD'
      DAY
12085 '                  'YY'
       YEAR
12090 '!!! THE YEARS 00-99 IN THE 20TH CENTURY ARE ASSU
      MED !!!
12100 GOSUB 12300 ' GREGORIAN TO JULIAN ENTRY, LOAD TAB
      LE
12110 IF MM<1 OR MM>12 OR DD<1 OR DD>31 OR YY<00 OR YY>
      99 THEN 12300
12120 IF DD=29 AND MM=2 AND INT(YY/4)*4=YY THEN 12140
12130 IF DD>VAL(MID$(VV$,MM*5-4,2)) THEN 12300  'ERROR
12140 V1=VAL(MID$(VV$,MM*5-2,3))+DD-1
12150 JD=YY*1000+V1
12160 IF INT(YY/4)*4=YY AND MM>2 THEN JD=JD+1
12170 RETURN
12180 PRINT"E R R O R":JD=0:GOTO 12170
12200 GOSUB 12300 ' JULIAN TO GREGORIAN ENTRY, LOAD TAB
      LE
12210 V4=0:V5=0:V7$=STR$(JD):IF LEN(V7$)<>6 THEN 12295
       'ERROR
12220 V6$=MID$(V7$,2,5):YY=VAL(MID$(V6$,1,2)):DD=VAL(MI
      D$(V6$,3,3))
12230 IF YY<00 OR OO<1 OR OO>366 THEN 12295 'ERROR
12235 IF INT(YY/4)*4 <> YY AND DD=366 THEN 12295 'ERROR
12240 IF INT(YY/4)*4=YY AND DD>59 THEN V4=1
12250 V5=V5+1
12260 IF DD=>VAL(MID$(VV$,V5*5-2,3))+V4 THEN 12250
12270 V3=VAL(MID$(VV$,V5*5-7,3)):IF V3>32 THEN V3=V3+V4
12280 DD=DD-V3+1:MM=V5-1
12290 RETURN
12295 PRINT"E R R O R":MM=0:DD=0:YY=0:GOTO 12290
12300 VV$="310031280323106030091311213015231102312133024
      43127430305313350036?":RETURN
```

Program Listing 1. Subroutine.

```
10 CLS:CLEAR 200
20 INPUT"ENTER GREGORIAN DATE (MM,DD,YY) ";MM,DD,YY
25 IF MM=0 THEN END
30 GOSUB 12100
40 IF JD=0 THEN 20
50 PRINTTAB(33) MM;"-";DD;"-";YY;"   IS  =";JD:PRINT
60 INPUT"ENTER JULIAN DATE (YYDDD) ";JD
70 GOSUB 12200
80 IF MM=0 THEN 60
90 PRINTTAB(27) JD;"   IS  =";MM;"-";DD;"-";YY:PRINT
100 GOTO 20
```

Program Listing 2. Driver.

Dear Subscriber:

80 Microcomputing does not keep subscription records on the premises. Therefore, calling the Peterborough offices doesn't solve your subscription problem.

To quickly solve your problem, please send your most recent address label and a description of the problem to:

80 Microcomputing
Subscription Department
PO Box 981
Farmingdale, NY 11737

Please allow the subscription department at least two weeks for an answer or a solution to your problem.

Thank you and enjoy your subscription.

Sincerely,

Debra L. Boudrieau

Debra L. Boudrieau
Circulation Manager

80 microcomputing™ bookshelf

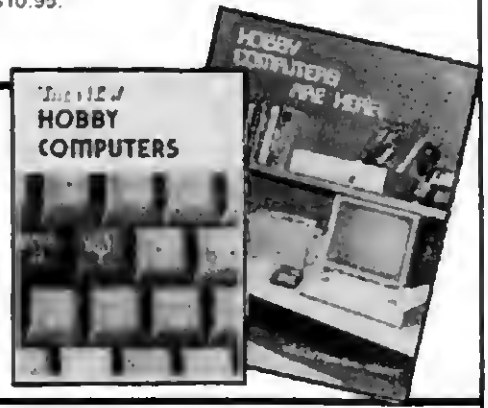
3 new books from the editors of KB & 80 Microcomputing

• **40 COMPUTER GAMES**—BK7381—Forty games in all in nine different categories. Games for large and small systems, and even a section on calculator games. Many versions of BASIC used and a wide variety of systems represented. A must for the serious computer gamesman. $7.95*

• **UNDERSTANDING AND PROGRAMMING MICROCOMPUTERS**—BK7382—A valuable addition to your computing library. This two part text includes the best articles that have appeared in 73 and Kilobaud Microcomputing magazines on the hardware and software aspects of the new microcomputing hobby. Well known authors and well structured text helps the reador get involved in America's fastest growing hobby. $10.95*

• **SOME OF THE BEST FROM KILOBAUD/MICROCOMPUTING**—BK7311—A collection of the best articles that have recently appeared in Kilobaud/MICROCOMPUTING. Included is material on the TRS-80 and PET systems, CP/M, the 8080/8085/Z80 chips, the ASR-33 terminal. Data base management, word processing, text editors and file structures are covered too. Programming techniques and hardcore hardware construction projects for modems, high speed cassette interfaces and TVTs are also included in this large format, 200 plus page edition. $10.95.*

INTRODUCTORY



• **THE NEW HOBBY COMPUTERS**—BK7340—This book takes it from where "HOBBY COMPUTERS ARE HERE!" leaves off, with chapters on Large Scale Integration, how to choose a microprocessor chip, an introduction to programming, low cost I/O for a computer, computer arithmetic, checking memory boards . . . and much, much more! Don't miss this tremendous value! Only $4.95.*

• **HOBBY COMPUTERS ARE HERE!**—BK7322—If you (or a friend) want to come up to speed on how computers work . . . hardware and software this is an excellent book. It starts with the fundamentals and explains the circuits and the basics of programming. This book has the highest recommendations as a teaching aid for newcomers. $4.95.*

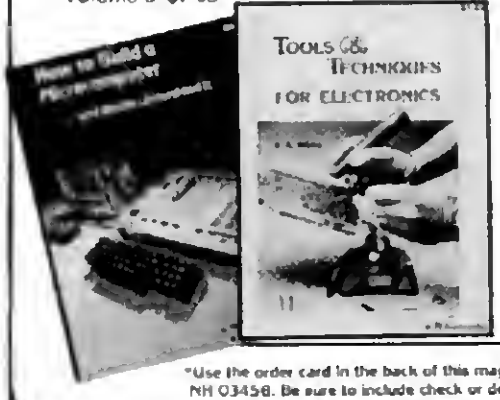
INTRODUCTION TO MICROCOMPUTERS (VOL. 0→III)

• **AN INTRODUCTION TO MICROCOMPUTERS, VOL. 0**—BK1130—The Beginner's Book—Written for readers who know nothing about computers—for those who have an interest in how to use computers—and for everyone else who must live with computers and should know a little about them. The first in a series of 4 volumes, this book will explain how computers work and what they can do. Computers have become an integral part of life and society. During any given day you are affected by computers, so start learning more about them with Volume 0. $7.95.*

• **VOL. I**—BK1030—**2nd Edition completely revised.** Dedicated to the basic concepts of microcomputers and hardware theory. The purpose of Volume I is to give you a thorough understanding of what microcomputers are. From basic concepts (which are covered in detail), Volume I builds the necessary components of a microcomputer system. This book highlights the difference between minicomputers and microcomputers. $12.50.*

• **VOL. II**—BK1040 (with binder)—$30.00*—Contains descriptions of individual microprocessors and support devices used only with the parent microprocessor. Volume II describes all available chips.

• **VOL. III**—BK1133 (with binder)—$20.00.* Contains descriptions of all support devices that can be used with any microprocessor.



• **HOW TO BUILD A MICROCOMPUTER—AND REALLY UNDERSTAND IT**—BK7325—by Sam Creason. The electronics hobbyist who wants to build his own microcomputer system now has a practical "How To" guidebook. This book is a combination technical manual and programming guide that takes the hobbyist step-by-step through the design, construction, testing and debugging of a complete microcomputer system. Must reading for anyone desiring a true understanding of small computer systems. $9.95.*

• **TOOLS & TECHNIQUES FOR ELECTRONICS**—BK7348—by A. A. Wicks is an easy-to-understand book written for the beginning kit builder as well as the experienced hobbyist. It has numerous pictures and descriptions of the safe and correct ways to use basic and specialized tools for electronic projects as well as specialized metal working tools and the chemical aids which are used in repair shops. $4.95.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

Unravel your ROM vacabulary.

Keyword List Plus

Jack Decker
1804 West 18th Street, Lot 155
Sault Ste. Marie, MI 49783

Maybe you've got all those BASIC keyword lists that show ASCII values coming out of your ears. (Two of them were published in issue 1 of *80 Microcomputing* alone.) But you probably can't put your hands on one when you want it.

Here, then, is a short program that prints the table and, as an extra bonus, the starting address for those keywords that are BASIC commands.

The BASIC keyword list can be divided into three parts—commands (such as GOTO, CLOAD, INPUT), functions (such as TO, AND, INT, MID$) and miscellaneous (TO, AND, ERR). All commands have a corresponding address in the jump table that starts at 6178 decimal and 5640 decimal respectively. The table contains the address bytes (least significant and most significant bytes) to which the computer should jump when the keyword is encountered.

The program lists the ASCII code representing the keyword, followed by the keyword itself. When applicable, additional columns show the LSB and MSB of the jump table entry, followed by that entry expressed in decimal (MSB•256 + LSB).

If you have always wanted to disassemble the ROMs, this program will point you to whatever BASIC command you want to tackle.

Line 35 is a delay loop to keep everything from scrolling too fast to read and may be changed or deleted.■

```
10 CLS:X=5712:A$="E":FORZ=1TO124
20 X=X+1:IFPEEK(X)<128THENA$=A$+CHR$(PEEK(X)):GOTO20
30 PRINTZ+127;A$,:IFZ<61A=6176+2*ZELSEIFZ>87ANDZ<124A=5
464+2*ZELSE35
32 B=PEEK(A):C=PEEK(A+1):PRINTB;C,B+C*256;
35 FORY=1TO100:NEXT
40 A$=CHR$(PEEK(X)-128):PRINT:NEXT
```

Program Listing

✓Reader Service—see page 274

80 microcomputing™ bookshelf

Z80 BOOKS

● **TRS-80 DISK AND OTHER MYSTERIES** – BK1181 – by Harvard C. Pennington. This is the definitive work on the TRS-80 disk system. It is full of detailed "How to" information with examples, samples and in-depth explanations suitable for beginners and professionals alike. The recovery of one lost file is worth the price alone. $22.50.*

● **PROGRAMMING THE Z-80** – BK1122 – by Rodnay Zaks. Here is assembly language programming for the Z-80 presented as a progressive, step-by-step course. This book is both an educational text *and* a self-contained reference book, useful to both the beginning and the experienced programmer who wish to learn about the Z-80. Exercises to test the reader are included. $14.95.*

● **Z-80 ASSEMBLY LANGUAGE PROGRAMMING** – BK1177 – by Lance A. Leventhal. This book thoroughly covers the Z80 instruction set, abounding in simple programming examples which illustrate software development concepts and actual assembly language usage. Features include Z80 I/O devices and interfacing methods, assembler conventions, and comparisons with 8080A/8085 instruction sets and interrupt structure. $16.99.*

● **Z-80 SOFTWARE GOURMET GUIDE AND COOKBOOK** – BK1045 – by Nat Wadsworth. Scelbi's newest cookbook! This book contains a complete description of the powerful Z-80 instruction set and a wide variety of programming information. Use the author's ingredients including routines, subroutines and short programs, choose a time-tested recipe and start cooking! $16.99.*

● **INTRODUCTION TO TRS-80 GRAPHICS** – BK1180 – by Don Inman. Dissatisfied with your Level I or Level II manual's coverage of graphics capabilities? This well-structured book (suitable for classroom use) is ideal for those who want to use all the graphics capabilities built into the TRS-80. A tutorial method is used with many demonstrations. It is based on the Level I, but all material is suitable for Level II use. $8.95.*

BASIC

● **BASIC BASIC (2ND EDITION)** – BK1026 – by James S. Coan. This is a textbook which incorporates the learning of computer programming using the BASIC language with the teaching of mathematics. Over 100 sample programs illustrate the techniques of the BASIC language and every section is followed by practical problems. This second edition covers character string handling and the use of data files. $9.45.*

● **LEARNING LEVEL II** – BK1175 – by David Lien. Written especially for the TRS-80, this book concentrates on Level II BASIC, exploring every important BASIC language capability. Updates are included for those who have studied the Level I User's Manual. Sections include: how to use the Editor, dual cassette operation, printers and peripheral devices, and the conversion of Level I programs to Level II. $15.95.*

● **THE BASIC HANDBOOK** – BK1174 – by David Lien. This book is unique. It is a virtual ENCYCLOPEDIA of BASIC. While not favoring one computer over another, it explains over 250 BASIC words, how to use them and alternate strategies. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word or combination of words. That's where the HANDBOOK comes in. It helps you get the most from your computer, be it a "bottom-of-the-line" micro or an oversized monster. $14.95.*

● **MY COMPUTER LIKES ME . . . WHEN I SPEAK BASIC** – BK1039 – An introduction to BASIC . . . simple enough for kids. If you want to teach BASIC to anyone quickly, this is the way to go. $3.95.*

● **ADVANCED BASIC** – BK1000 – Applications, including strings and files, coordinate geometry, area, sequences and series, simulation, graphing and games. $9.65*.

● **SIXTY CHALLENGING PROBLEMS WITH BASIC SOLUTIONS (2nd Edition)** – BK1073 – by Donald Spencer, provides the serious student of BASIC programming with interesting problems and solutions. No knowledge of math above algebra required. Includes a number of game programs, as well as programs for financial interest, conversions and numeric manipulations. $6.95.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf ● Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

80 microcomputing™ bookshelf

GAMES

NEW!

• MORE BASIC COMPUTER GAMES – BK1182 – edited by David H. Ahl. More fun in BASIC! 84 new games from the people who brought you *BASIC Computer Games*. Includes such favorites as Minotaur (battle the mythical beast) and Eliza (unload your troubles on the doctor at bargain rates). Complete with game description, listing and sample run. $7.50.*

• WHAT TO DO AFTER YOU HIT RETURN – BK1071 – PCC's first book of computer games 48 different computer games you can play in BASIC programs, descriptions, many illustrations. Lunar Landing, Hammurabi, King, Civet 2, Qubic 5, Taxman, Star Trek, Crash, Market, etc. $10.95.*

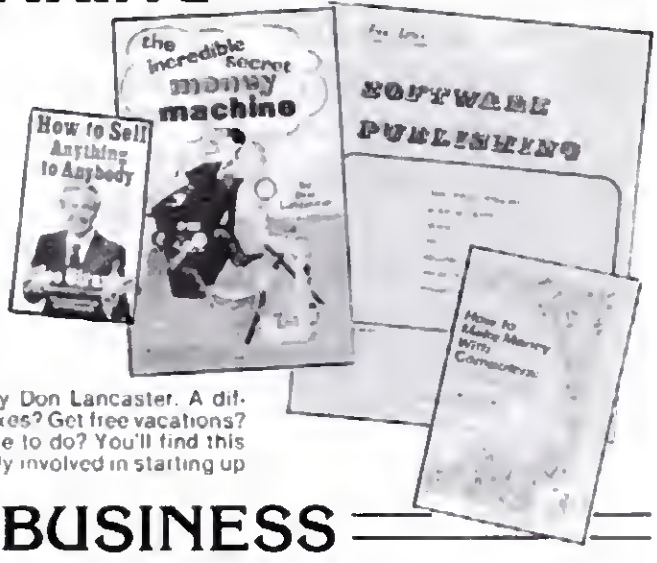
• BASIC COMPUTER GAMES – BK1074 – Okay, so once you get your computer and are running in BASIC, then what? Then you need some programs in BASIC, that's what. This book has 101 games for you from very simple to real buggers. You get the games, a description of the games, the listing to put in your computer and a sample run to show you how they work. Fun. Any one game will be worth more than the price of the book for the fun you and your family will have with it. $7.50.*

MONEY-MAKING

• HOW TO MAKE MONEY WITH COMPUTERS – BK1003 – In 10 information-packed chapters, Jerry Felsen describes more than 30 computer-related, money-making, high profit, low capital investment opportunities. $15.00.*

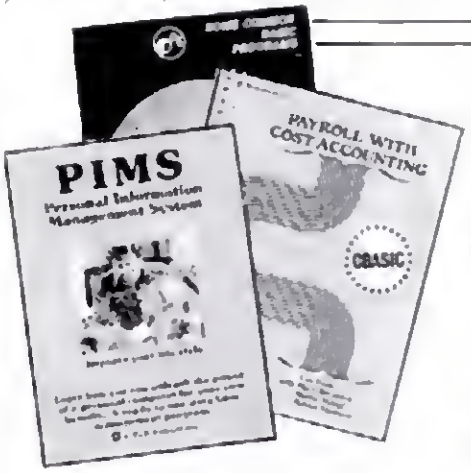
• HOW TO SELL ANYTHING TO ANYBODY – BK7306 – According to *The Guinness Book of World Records*, the author, Joe Girard, is "the world's greatest salesman." This book reveals how he made a fortune and how you can, too. $2.25.*

• FREELANCE SOFTWARE PUBLISHING – BK1179 – by B. J. Korites. "This book is about money and how to make it by writing and selling computer programs," (author's foreword) If you have the skills to write a saleable program, you now need to acquire the skills to sell that program. This compact book comprehensively covers the entire publishing process and many aspects of software salesmanship. $14.95.*



• THE INCREDIBLE SECRET MONEY MACHINE – BK1178 – by Don Lancaster. A different kind of "cookbook" from Don Lancaster. Want to slash taxes? Get free vacations? Win at investments? Make money from something that you *like* to do? You'll find this book essential to give you the key insider details of what is really involved in starting up your own money machine. $5.95.*

BUSINESS



• PAYROLL WITH COST ACCOUNTING – IN BASIC – BK1001 – by L. Poole & M. Borchers, includes program listings with remarks, descriptions, discussions of the principle behind each program, file layouts, and a complete user's manual with step-by-step instructions, flowcharts, and simple reports and CRT displays. Payroll and cost accounting features include separate payrolls for up to 10 companies, time-tested interactive data entry, easy correction of data entry errors, job costing (labor of distribution), check printing with full deduction and pay detail, and 16 different printed reports, including W-2 and 941 (in CBASIC) $20.00.*

• SOME COMMON BASIC PROGRAMS – BK1053 – published by Adam Osborne & Associates, Inc. Perfect for non-technical computerists requiring ready-to-use programs. Business programs, plus miscellaneous programs. Invaluable for the user who is not an experienced programmer. All will operate in the stand-alone mode. $12.50 paperback.*

• PIMS: PERSONAL INFORMATION MANAGEMENT SYSTEM – BK1009 – Learn how to unleash the power of a personal computer for your own benefit in this ready-to-use data-base management program. $11.95.*

*Use the order card in the back of this magazine or itemize your order on a separate piece of paper and mail to 80 Microcomputing Bookshelf • Peterborough NH 03458. Be sure to include check or detailed credit card information. No C.O.D. orders accepted. All above add $1.00 handling. Please allow 4–6 weeks for delivery. Questions regarding your order? Please write Customer Service at the above address.

FOR TOLL FREE ORDERING CALL 1-800-258-5473

HOBBYWORLD® ELECTRONICS, INC.

19511 BUSINESS CENTER DRIVE, DEPT. V1
NORTHRIDGE, CALIFORNIA 91324

Call Toll-Free: USA (800) 423-5387
In California: (800) 382-3651
Local & Outside USA: (213) 886-9200

✓23

TELESIS VAR/80 I/O Unit
for the TRS-80

Now you can use your TRS-80 as a digital door lock, burglar alarm, power manager, frequency counter, light dimmer, darkroom timer, just to name a few! Comes fully assembled and tested. Use it with or without an expansion interface. Data pack includes: instructions, applications, sample circuits and several programs. With power supply. Wt. 5 lbs.

Cat. No. 1092 **$105**

Mini 8100 S-100 Bus Adaptor
for the TRS-80

Mini size, mini price, maxi performance! A complete adaptor/motherboard for the TRS-80

Cat No.	Description	Price
1905	Kit, all parts, one S-100 connector	$115.45
1906	A&T w/four S-100 connectors	$155.45
1907	Kit, w/5-100 sz bd, plugs into mainframe	$ 90.00
1908	As above, a&t	$125.00

16K MEMORY ADD-ON
for the TRS-80, Apple, & Exidy $39.95

Everything you need to upgrade your system! Includes 4 pages of illustrated instructions. Complete with RAMS and pre-programmed jumpers. No special tools required! Wt 4 oz.

Cat No.	Description
1156	For TRS-80 keyboard unit
1156A	For TRS-80 exp. interface purchased before 4/1/79
1156B	For TRS-80 exp. interface purchased after 4/1/79
1156C	APPLE II
1156D	EXIDY

BASF 5¼" DISKETTES $35 Box/10

Soft sector, double density, single sided. Use for TRS-80, Apple, Atari.
Cat No. 2746 Box of 10 diskettes.

VERBATIM 5¼" Diskettes

VERBATIM 525 SERIES
- Double Density
- Single sided
- Perfect for commercial and general applications

Cat No.	Description	Type	Use for	10 for
1147	Soft sector	525-01	TRS-80, Apple	$33.00
1148	10 hole, hard	525-10	North Star	33.00
1149	16 hole, hard	525-16	Micropolis	33.00

VERBATIM 550 SERIES
- Quad Density (double sided, double density)
- For commercial and general applications

1492	Soft sector	550-01	SA650, MP152	$62.25
2328	10 hole, hard	550-10	BASF, Wangco	51.95
2529	16 hole, hard	550-16	Micro-2	59.95

VERBATIM 577 SERIES
- Certified twice, 77 tracks
- Single sided, double density
- Built-in hub protector ring
- For critical data applications

2550	Soft sector	577-01	TRS-80, Apple	$49.95
2331	10 hole, hard	577-10	North Star	54.93
2332	16 hole, hard	577-16	Micropolis, etc.	49.95

Electric Pencil for the TRS-80®

Allows you to produce mailing-lists, forms, large numbers of original correspondence, etc. A character-oriented word processing system, providing maximum freedom and simplicity in the handing of text. Eliminates the need for word hyphenations or carriage returns. Line formatting is done automatically. Insert, delete, or relocate any text using simple keyboard demands.

Cat No. 1338 TRS-80, L1 & L2 16K, Cassette **$59.95**

Cat No. 1338-D TRS-80, L1 & L2 16K, Disk **$89.95**

"Scott Adams" ADVENTURES

ADVENTURELAND

Cat No.	Description	Price
2719	TRS-80, L2, 16k Cassette	$14.95
2720	TRS-80, L2, 32k Disk (+ Pirates Adv.)	$24.95

PIRATES ADVENTURE

2505	TRS-80, L2, 16k Cassette	$18.95

MISSION IMPOSSIBLE

2725	TRS-80, L2, 16k Cassette	$14.95
2724	TRS-80, L2, 32k Disk (+ Voodoo Castle)	$26.95

The COUNT

2726	TRS-80, L2, 16k Cassette	$14.95

STRANGE ODYSSEY

2766	TRS-80, L2, 16k Cassette	$14.95

GHOST TOWN

2765	TRS-80, L2, 16k Cassette	$14.95

TUNNEL OF FAHAD

2771	TRS-80, L2, 16k Cassette	$ 9.95

Utilities, Business, Sci-fi, Games, Education, and much more software for the TRS-80 is available at HobbyWorld!

Disk/Diskette Drive Head Cleaning Kit

Clean these hard-to-reach heads in just minutes! Available for both 5¼" and 8" drives, single and double sided. Comes complete with two cleaning disks, 4 oz. of CS-85 cleaning solution, and easy-pour dispenser. Wt. 12 oz.

Cat No. 2499 8" disk
Cat No. 2534 5¼" diskette **$30**

FREE CATALOG

Fourty-four pages of computers, terminals, printers, disk-drives and many more peripherals that can add dimension to your personal computing. We also carry complete lines of computerized toys & games, application boards, integrated circuits, comprehensive software, electronics parts, D.C. & soldering accessories, electronics books & manuals, and much more!

Call or write to us today and ask us for your free HobbyWorld catalog. (or circle the reader's service number in this magazine.)

HOW TO ORDER

Minimum Order $15.00. Order toll-free by phone or by mail, or at our retail stores. Pay by check, Mastercard, Visa or C.O.D. Please include expiration date with charge-card orders. U.S. dollars only. Include phone # and magazine issue you are ordering from. Add $1.25 for C.O.D. and shipping charges from rates below.

Shipping Rates: USA
Ground: $2.25 for first 2 lbs. and 40¢ each addt'l lb.
Air: $3.25 for first 2 lbs. and 70¢ each addt'l lb.

Shipping Rates: Foreign
Ground: $3.00 first 2 lbs. and 60¢ each addt'l lb.
Air: $11.25 first 2 lbs. and $5.00 each addt'l lb.

Prices valid thru month of magazine issue. Some items subject to prior sale or quantity limits. HobbyWorld is not responsible for typographical errors. 120 day Guaranteed Satisfaction. Exception: Partially assembled kits, abuse or misuse.

TRS-80 SERIAL I/O

• Can input into basic • Can use LIST and LPRINT to output, or output continuously • RS-232 compatible • Can be used with or without the expansion bus • On board switch selectable baud rates of 110, 150, 300, 600, 1200, 2400, parity or no parity odd or even, 5 to 8 data bits, and 1 or 2 stop bits O T R. line • Requires +5, -12 VDC • Board only $19.95 Part No. 8010, with parts $59.95 Part No. 8010A, assembled $79.95 Part No 8010 C. No connectors provided, see below.

EIA/RS-232 connector Part No DB25P $6.00, with 9' 8 conductor cable #10.95 Part No DB25P9

3' ribbon cable with attached connectors to fit TRS-80 and our serial board $19.95 Part No. 3CAB40

COMPUCRUISE

$129.95; with cruise control $169.95

PAPER TIGER

Prints address labels, multicopy invoices and legal-size reports. Adjust the tractor width from 1-3/4 to 9-1/2 inches. 8 switch-selectable forms lengths. Print 6 or 8 lines per inch. Add the software-selectable full dot plotting graphics option to print illustrations, block letters, charts, graphs. Part No. 162172 $699.95 • with graphics option Part No. 162173 $1099.95

GAME PADDLES & SOUND

Includes: 2 game paddles, interface, software, speaker, power supply, full documentation including: schematics, theory of operation, and user guide; plus 2 games on cassette (Pong and Starship War). $79.95 Complete Part No. 7922C

DIGICOM DATA PRODUCTS INC. Series 312 Acoustic Coupler

300 BAUD Originate, Part No. AC3122, $219.95. 300 BAUD Answer, Part No AC3122, $219.95. 300 BAUD Answer/Originate, Part No. AC3123, $229.95.

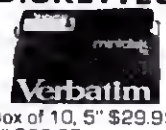
IBEX LIGHT PEN

Comes with Backgammon and Tic-Tac-Toe on tape with full documentation and program listing. Requires 9v. battery. Part No. IBEX $19.95

SYSTEM EXPANSION from LNW Research

• Serial RS232C/20 mA I/O • Floppy controller • 32K bytes memory • Parallel printer port • Dual cassette port • Real-time clock • Screen printer bus • Onboard power supply • Software compatible • Solder mask, silk screen. PC board and user manual. Part No. LNW80, $69.95.

DISKETTES



Box of 10, 5" $29.95, 8" $39.95.
Plastic box, holds 10 diskettes, 5" - $4.50, 8" - $6.50.

16K RAMS

For the Apple, TRS-80 or Pet $8 each Part No. 4116/2117

LEEDEX MONITOR

12" Black and White • 12 MHz Bandwidth • Handsome Plastic Case • $139.00

S-100 INTERFACE

AN S-100 bus Adapter—Motherboard for the TRS-80. Kit, Part No. HUHB10LXK, $295.95. Assembeled, Part No. HUHB10LXA, $375.95.

NOW! A FULL SUPPORT SYSTEM FOR TRS-80

• 32K of RAM • EPROM firmware • Disk control • Data acquisition • Parallel I/O • Serial I/O • Plug into GPA's Motherboard. GPA's quality design includes • 6-44 pin edge connectors • +5V, -5V, +12V, -12V external power supply required • Active termination. The Motherboard, Part No. GPA80, is only $149.95.

TAKE ADVANTAGE OF GPA-EXPANSION CARDS FOR THE GPA80

Memory cards: Now with Fortran compilers available for your TRS-80, additional expansion memory is a must! Card with sockets only, Part No. GPA801, $119.95. Card with 16K of 4116 Dynamic Ram, Part No. GPA802, $224.95. Card with 32K of 4116 Dynamic Ram, Part No. GPA803, $329.95. All cards come equipped with sockets to accomodate 32K of Ram.

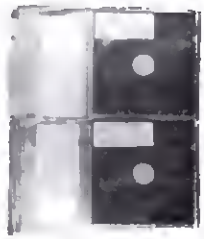
EPROM firmware card. Put those valuable subroutines in firmware. Don't waste time loading and unloading tapes and disks. For 2708 or 2716 EPROMS, Part No. GPA806, $79.95.

Serial I/O card. Here's what you've been asking for, a full serial terminal interface, with RS-232C or 20 mA. Current loop. Input/output capabilities. Part No. GPA807, $79.95.

Parallel I/O Card. Control functions in the outside world, monitor and store real time events. Two parallel output ports. Dip switches select ports (0-254). Part No. GPA808, $79.95.

FLOPPY DISK STORAGE BINDERS

Three ring binder comes with ten transparent plastic sleeves which accommodate either twenty, five-inch or ten, eight-inch floppy disks. Binder & 10 holders, Part No. B108—$9.95 • Extra holders, Part No. B10—69¢ each.



Three-ring binder with ten 5 1/4 inch jackets Part No. 5108—$9.95• Jackets only. fits standard 3-ring binders, Part No. 510—69¢ each.

DIGITAL CASSETTE

5 min. each side. Box of 10 $9.95. Part No. C-5.

TRENDCOM PRINTER

TRENDCOM 200, Part No. TRC0200 $495.95. **Interface** for TRS-80, Part No. T80A $49.95. For Apple II, Part No. TRCALL, $75.95. For PET, NO. TRCP2, $79.95. For Scoccerer, TRCSR1 $45.95.

SARGON: A Computer Chess Program

Features the complete program that won the 1978 West Coast Computer Faire Tournament. Part No. 00603 — TRS-80 Level II; Part No. 00604 — Apple II (24K). $19.95

SPINWRITER MODELS 5510 and 5520

Features—EIA RS-232C/CCITT V.24 Interface Standard • 55 Characters Per Second Maximum Print Rate • Impeccable Print Quality (OCR Quality) • Microprocessor Electronics • High Resolution Plotting/Graphing • Lowest Operating Noise Level • Self-Test Printing • Operator Engineered Control Panel • Prints Original and up to Seven Copies • NEC Information Systems new Model 5510 Receive Only and Model 5520 Keyboard Send/Receive SPINWRITER terminals are microprocessor controlled serial, impact terminals designed for remote printing applications where impeccable print quality is required. Model 5510 RO, Part No. NECA30759 $2795.95 • Model 5520 KSR, Part No. NECA30762 $3095.95

Send for FREE Catalog...a big self addressed envelope with 80¢ postage gets it fastest!

To Order: Mention part no., description, and price. In USA shipping paid by us for orders accompanied by check or money order. We accept C.O.D. orders (U.S. only) or a VISA or Master Charge no., expiration date, signature and phone no., shipping charges will be added. CA residents add 6.5% for tax. Outside USA add 15% for air mail postage and handling. Payment must be in U.S. dollars. Dealer inquiries invited. Prices subject to change without notice.

VISA

26

Order Line: (408) 448-0800

ELECTRONIC SYSTEMS Dept. 80, P.O. Box 21638, San Jose, CA USA 95151

Reader Service—see page 274

1980 Article Index for *80 Microcomputing*

LOWEST PRICES ON PERSONAL COMPUTERS

buy by mail and save

apple computer

Apple II personal computer.

16K
List $1195

ONLY $989

32K, List $1395 ..... $1169
48K, List $1259

DISK II DRIVE $420
Above w/Cantroller $505
MICROSOFT Z80/CPM
Conversion For Apple II ... ONLY $299

APPLE III
w/96K $2998

CENTRONICS PRINTER INTERFACE
Pascal Language System List $495 **$420**
Centronics
Printer Card List $225 ..... **$191**
High Speed
Printer Interface $195 ..... **$165**

COMPLETE LINE OF CALIFORNIA COMPUTERS
Interface cards available.
We also stock the
DC Hayes Micromodem, Mountain Hardware, and the **SSM combination serial/parallel interfaces.**

Personal Computer Systems ~24

609 Butternut St.,
Syracuse, NY 13208
(315) 475-6800

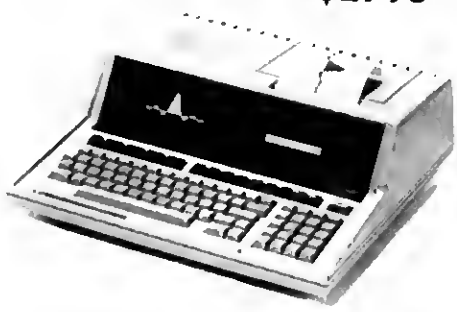
VISA

master charge

Prices do not include shipping by UPS. All prices and offers subject to change without notice.

HEWLETT PACKARD

HP-85A ONLY $2795



HP-85 ACCESSORIES

5-¼" Dual Master Disc Drive List $2500 ..... **$2125**
5-¼" Single Master Disc Drive List $1500 ..... **$1275**
HP 7225A Graphics Platter List $2050 **$1845**
HP-85 16K Memary Madule List $395 . **$355**
HP-85 Application Pacs Standard List $95 ..... **$85**
Seriol (RS-232C) Interface Module List $395 **$355**
GPIO Interface Madule List $495 **$445**

IMAGINE A CALCULATOR YOU CAN CUSTOMIZE. IT'S HERE—THE HP-41C.

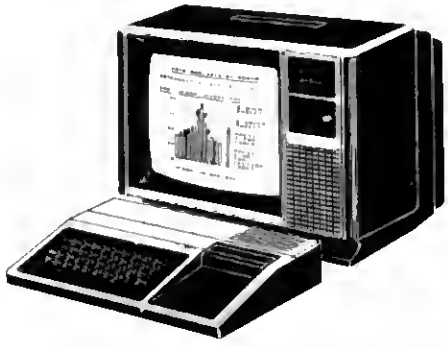
HP-41C
ONLY $244.95

CALCULATORS:

HP-32E Scientific w/Statistics	$3.95
HP-33C Scientific Programmable	99.95
HP-34C Advanced Scientific Programmable	123.95
HP-37E Business Calculator	58.95
HP-67 Handheld Fully Advanced Programmable Scientific for Business & Engineering	298.95
HP-97 Desktop w/Built-in Printer	579.95

TEXAS INSTRUMENTS INCORPORATED

TI-99/4
home computer



CALL FOR PRICE

Commodore Pet
CALL FOR PRICE

PRINTERS FOR ALL COMPUTERS

ATARI

A Warner Communications Company

PERSONAL COMPUTER SYSTEMS

List $1080
ATARI® 800™ PERSONAL COMPUTER SYSTEM. $849

ATARI® 810 DISC DRIVE
List $699.95 ..... **$589**

ATARI®
820™ Printer, List $599.95 .... **$499**
Atari® 400 List $630 .... **$499**

80 Reviews

Books

PRODUCT	MANUFACTURER	ISSUE
1881 THINGS W/PERSONAL COMPUT.	TAB BOOKS	DEC/81
88 PROGAMES FOR THE TRS-80	WAYNE GREEN, INC.	APR/80
80-US	80 AM PUBLISHING	FEB/81
AN INTRO. TO COMPUTER MUSIC	JOHN WILEY & SONS, INC.	DEC/80
COMPUTER GAMES FOR BUS.& SCHL.	WINTHROP PUBLISHERS, INC.	SEP/81
FREELANCE SOFTWARE PUBLISHING	REMS PUBLICATIONS	FEB/81
GUIDE TO TRS-80 INFORMATION	F.E. HOHENER	FEB/80
INSIDE LEVEL II	MUMFORD MICRO SYSTEMS	OCT/80
INTRO. TO TRS-80 GRAPHICS	DILITHIUM PRESS	APR/81
INTRODUCTION TO T-80C	DILITHIUM PRESS	JUL/81
LEARNING LEVEL II	COMPUSOFT PUBLISHING	MAY/81
MICRO MILLENIUM, THE	WIRING PRESS	AUG/81
MOST POP. SUBROUTINES IN BASIC	TAB BOOKS	NOV/81
PASCAL-INTRO TO LOGICAL PGMNG.	COMPUTER SCIENCE PRESS	MAY/80
PER. GUIDE FOR COMPUTERISTS	E. BERG	FEB/81
PROB SOLVING/STRUCTURED PGMNG.	ADDISON-WELESLEY	MAY/81
PROGM TECHS FOR LEVEL1 BASIC	TANDY/RADIO SHACK	NOV/81
PROGRAMMERS BOOK OF RULES, THE	LIFETIME LEARNING PUBLICATIONS	MAY/81
RECREATIONAL COMPUTING	PEOPLES COMPUTER CO.	FEB/80
RUNNING WILD,THE NEXT IND. REV	OSBORNE/MCGRAW HILL	APR/81
SOFTWARE BUYERS GUIDE	WALLACE ELECTRONICS	FEB/80
SSI MICRO SOFTWARE GUIDE, THE	SSI	FEB/81
SUPERMAP	FULLER SOFTWARE	JUL/80
TRS-80 DISASSEMBLED HANDBOOK	MICROCRAFT ENGINEERING	JUN/80
TRS-80 INTERFACING	BLACKSBURG EDUCATION SERIES	SEP/80
TRS-80 MONTHLY NEWSLETTER	MJE COMPUTRONICS	FEB/80
TRS-80 TECHNICAL REFERENCE MAN	TANDY/RADIO SHACK	MAY/80

Hardware

PRODUCT	MANUFACTURER	ISSUE
ACU-DATA TAPE DIGITIZER	ALPHASETICS MFG.	JUN/80
ANADEX PRINTER	ANADEX	FEB/80
BETA-80	RECA	MAY/80
CENTRONICS 730 PRINTER	CENTRONICS INC.	JUL/80
COMPRINT 932	COMPUTSE PRINTERS INTERNATIONAL	NOV/80
DB-9500 LINE PRINTER	ANADEX, INC.	OCT/80
EXATRON STRINGY FLOPPY	EXATRON	MAY/80
HIGH SPEED MODIFICATION KIT	SIMUTEK	NOV/80
MAYDAY +8	SUN-RESEARCH INC.	JUN/80
MICROLINE-80	OKIDATA	OCT/80
MODEL 448 PAPER TIGER	INTEGRAL DATA SYSTEMS, INC.	OCT/80
MODEL 888 PRINTER	BASE-2	APR/80
PERCOM DISK DRIVES	PERCOM	FEB/81
QUICK PRINTER	CENTRONICS DATA COMPUTER CORP.	MAR/80
RS-232 BOARD	TANDY/RADIO SHACK	MAR/80
TC-8 CASSETTE SYSTEM	JPC PRODUCTS	JUN/80
TRENDCOM 188 PRINTER	TRENDCOM	AUG/80
TRS-80	TANDY/RADIO SHACK	AUG/80
TRS-80 MODEL II	TANDY/RADIO SHACK	JUL/80
TRS-80 VOICE SYNTHESIZER	TANDY/RADIO SHACK	APR/80

Software

PRODUCT	MANUFACTURER	ISSUE
ADVENTURE	SOFTWIN ASSOC/MICROSOFT	JUN/81
AIR FLIGHT SIMULATION	INSTANT SOFTWARE INC.	APR/81
ANDROID NIM	80-MW PUBLISHING CO.	FEB/80
APPLICATIONS	DILITHIUM TAPES	APR/80
AUTOK AND QEDIT	DISCOVERY BAY SOFTWARE	FEB/81
BASIC 1P	SMALL SYSTEMS SOFTWARE	MAY/81
BOOTSTRAP	PRACTICAL APPLICATIONS	JUL/81
BUSINESS ANAL SYSTEM	THE BOTTOM SHELF	FEB/81
C BASIC	PNC CORP.	APR/81
DISK DIRECTORY	MUMFORD MICROSYSTEMS	JAN/81
DISK INSTRUCTION COURSE	TANDY/RADIO SHACK	NOV/80
DISK*MOD	MISOSYS	DEC/81
DOCTOR AND FETCH	ORICSON SOFTWARE	JAN/81
EDAS 4.8	GALACTIC SOFTWARE	AUG/80
EDUCATIONAL	DILITHIUM TAPES	APR/80
ESP-1	SMALL SYSTEMS SOFTWARE	FEB/81
GAMES	DILITHIUM TAPES	APR/80
GENERALIZED SUBROUTINE FACILTY	RACET COMPUTES	JAN/81
GOMOKU	DISCOVERY BAY SOFTWARE	FEB/81
GRAPHICS AND MISCELLANEOUS	DILITHIUM TAPES	APR/80
ION-IV DATA BASE MANAGER	MICRO ARCHITECT	NOV/80
INDIVIDUAL STUDY CENTER	TTC SOFTWARE	APR/80
INSEQ-80 AND INSORT-80	S & M SYSTEMS, INC.	OCT/80
INTERLUDE -ULTIMATE EXPERIENCE	SYNTONIC SOFTWARE	SEP/80
ISAM	JOHNSON ASSOCIATES	SEP/80
ISAM,INFO, STORAGE & RETRIEVAL	THE ALTERNATE SOURCE (TAS)	DEC/80
KEEPIT VERSION 2.0	THE ALTERNATE SOURCE (TAS)	DEC/80
LEVEL I IN LEVEL II	APPHAZE SOFTWARE	MAR/80
LEVEL III	MICROSOFT	ARY/80
LINE RENUMBERING	SOFTWARE ASSOCIATION	JAN/80
MAIL/FILE LIST	GALACTIC SOFTWARE	FEB/80
MAILLIST	DAR SALES	OCT/80
MATHEMATICS	DILITHIUM TAPES	APR/80
MAXI-DISK AND SHUFFLEBOARD	PARASITIC ENG.	ARY/80
MICRO MUSIC	TANDY/RADIO SHACK	OCT/80
MICRO-OPOLY	LEVEL IV PRODUCTS, INC.	OCT/81
MMS FORTH	MILLER MICROCOMPUTING SERVICAS	NOV/80
MON-2	HUBERT HOWE	FEB/80
MORSE CODE XMIT & REC PGM.	MICROCRAFT ENGINEERING	JUL/80
MOVING SIGNBOARD	CIRCLE ENTERPRISES	FEB/81
MUSIC COMPOSER/EDITOR	PFDC SOFTWARE	JAN/81
NAME AND ADDRESS SYSTEM	SMALL BUSINESS SYSTEMS GROUP	FEB/81
OBJNEL	HUBERT HOWE	FEB/81
PEOPLE'S PASCAL I AND II	COMPUTER INFORMATION EXCHANGE	ARY/81
PLANETARY LANDER	INSTANT SOFTWARE INC.	SEP/81
POOR MAN'S TEXT EDITOR	DON COOK	FEB/81
PTMAN1D	TANDY/RADIO SHACK	AUG/81
RADEX-1B	IJG COMPUTER SERVICES DIV.	JUL/80
RADIO SHACK MAILER	TANDY/RADIO SHACK	FEB/80
REMODLE-PADLOAD	RACET COMPUTES	MAR/80
REMUMBER	TANDY/RADIO SHACK	MAR/80
RSM-2 MONITOR	SMALL SYSTEMS SOFTWARE	APR/81
SCRIPSIT	TANDY/RADIO SHACK	MAR/81
SORT-II	NORTHEAST MICROWAVE	FEB/80
SPECIAL DELIVERY	SOFTWARE ETC.	JUL/80
STEP BY STEP..	PROGRAM DESIGN INC.	FEB/80
T-SHORT	WEB ASSOCIATESD	APR/80
TRS 232 FORMATTER	SMALL SYSTEMS SOFTWARE	OCT/80
VIDEO CHECKERS	COMPU-QUOTE	FEB/81
WIN 21	DISCOVERY BAY SOFTWARE	FEB/80
WORD-1	MICRO ARCHITECT	JUN/80
WORD-IV--DISK WORD PROCESSOR	MICRO ARCHITECT	NOV/80
188 IAP/CMD	ORG-TEX INDUSTRIES	SEP/80

2:80	Disk database	*Floppy PIMS*	Herman
2:97	Four games reviewed	*Games Review*	Hallen
2:111	Cheap text editor	*Poor Man's Text Editor*	Blechman
3:58	Three programming aids	*Useful Utilities*	Leedham
3:77	Private label compared	*Quick Printer*	Riekers
3:134	Level I in a Level II	*One into Two*	Wantz
3:136	Radio Shack's interface	*RS232*	Hicks
4:70	Small systems software	*RSM-2 Monitor*	Churchill
4:130	CBASIC from FMG	*BASIC Review*	Knecht
4:136	Software from dillithium	*Dillithium Tapes*	Hallen
5:38	Computer information exchange	*Pascal I&II*	Monsour
5:56	Word processors compared	*Pencil vs. Scripsit*	Perry
5:58	Extra commands with Level III	*Level III*	Bobo
5:82	Stringy floppy and BETA-80	*Disk Alternatives*	Dyk
6:92	Six programs from four companies	*Applications Software*	Hallen
7:100	Morse code	*Software for Hams*	Richardson
7:124	Printer	*Centronics 730*	Frankenberg
8:184	Operation	*The TRS-80*	MacLean
9:154	Voice synthesizer	*Eloquent Eighties*	Wright
11:125	RS pocket computer	*BASIC in the Palm of Your Hand*	Knecht
12:102	Statistical programs	*STATS*	Johnson

UTILITIES

1:68	Blinking cursor subroutine	*Winking Cursor*	Lovy
1:82	Renumber BASIC	*Basic BASIC Renumbering*	Orleff
1:118	Relocating T-Bug	*Get T-Bug High*	Rappaport
1:120	Put EDTASM on DOS	*EDTASM on Disk*	Butler
1:122	Tape analysis program	*TTape*	Stevens
1:134	Increasing variables	*Extra Variables*	Clark
2:42	Machine language monitor	*BABYBUG*	Kitsz
2:68	Modify your monitor	*CLOAD Machine Language*	Schimelman
2:82	Add BASIC statements	*APPEND It!*	Gerald
2:118	Program shrinker	*Compress It!*	Powers
2:120	Format printouts	*LPRINT Formatter*	McCormick
2:124	Index program names	*Disk Index*	Gheshire
3:46	Print files while running	*SPOOL & DeSPOOL*	Gentry
3:80	Memory test	*Test your memory.*	Chepko
3:105	Hard copy video	*LPRINT Routines*	Werner
3:115	Identify system and BASIC tapes	*Whazit?*	Penny
3:122	Simple text editor	*Screen Editor*	Colsher
3:125	Error messages in Level II	*Extra Errors*	Moses
4:62	CLOAD assembly programs	*Level II to Level I*	Wolf
4:68	Adding sound effects	*Babybeep*	Kitsz
4:80	Adding USR subroutines	*Multiple USRs*	Ventimiglia
4:84	Shortening T-BUG	*T-BUG for II*	Curtis
4:90	Putting machine-language into Level II	*MACROPOKE Monitor*	Suter
4:106	Consolidating SYSTEM programs	*Service Tape*	Flatley
4:108	Relocating KBFIX	*KBFIX Fix*	Andreesen
4:133	Designing an intelligent terminal	*BASIC Terminal*	Noreault
5:76	Finding defective memory locations	*Babyroot*	Kitsz
5:86	Determining quality of input	*CLOAD Monitor*	Whaland
5:96	ASCII and hexcodes on screen	*Backup/Display*	Lindley
5:114	Speed up DOS	*FASTDOS*	Neher
5:116	Designing and utilizing video layouts	*Etch-a-Screen*	Shrum
5:126	Editing hybrid programs	*Progdata*	Kelley
5:130	Producing sound through recorders	*Super Sound*	Morr
5:134	Block movement	*Cutting and Splicing BASIC*	Nottingham
5:136	Printing the screen	*LPVIDEO*	Powers
6:88	Displaying hex conversions	*Hex Display*	Campbell
6:111	DECwriter LA-34	*DECwriter Driver*	Beauchamp
6:132	Displaying buffer contents	*Buffer Analysis*	Chambers
6:134	Displaying lots of data	*Display Formatting*	Joffe
6:146	Treat assembly like BASIC	*CLOAD Assembly Language*	Baker
7:136	Sound generation	*Sound X*	Baker
7:158	Backup copy	*Displaced Programs*	Moehlis
7:160	Tape duplication	*TCOPY*	Stevens
7:162	TRENDCOM 100	*FORMAT 40*	Adams
8:100	Index	*Tape Librarian*	Herold
8:107	Security	*The Invisible Password*	Conley
8:108	Telling zeros	*Slash Zero*	Richardson
8:110	Cryptology	*Code Cracker*	Morgan
8:118	Security	*Software Lock*	Kelleher
8:121	Lowercase	*Lowercase with Strings Attached*	Chepko
8:122	Modifying EDTASM	*Custom EDTASM*	Blair
8:132	Combining machine and BASIC	*AUTOPOKE*	Kump
8:160	Worksheet	*The Graphics Coder*	Racine
8:164	Sequential file	*Disk File Protection*	Keen and Dischert
8:170	Electric Pencil	*Pencil RS232 Driver*	Kinsey
8:174	Tape index	*Cassette File*	Tallman
9:68	Deleting spaces	*Free Space*	Cornell
9:76	Keywords	*Uni-key*	Archer
9:88	Variable names	*Document Those Variables*	Noel
9:94	Centronics 779	*Printer Calibration*	Rexrode
9:98	INPUT	*Versatile Input*	Wilde
9:146	Index program	*Reference Library Index*	Morgan
9:150	POSDIS	*Position Display*	Frost
9:168	Delaying a program	*Delay Loop*	Joffe
9:170	Rectangles, ellipses, boxes	*Divine Proportions*	Cecil
9:173	Moving messages	*Walking Words*	Borrmann
9:192	Sort utility	*Beyond Shell Metzner*	Walker
9:196	Debug monitor	*Deflower Your Debug*	Walter
9:202	Scrolling	*Slow Scroll*	Lewis
9:206	Disk operation	*OWIKOISK*	Nazarian
9:210	Cursor	*The Competition's Cursor*	Bishop
10:134	USR	*Variable Scroll*	Colsher
10:138	INKEY$	*Input with Insight*	Decker
10:202	Commands	*Super Graphics*	Moyer
10:207	Multiple loading into memory	*Triple Play for T-BUG*	Johnson
10:210	Maxell UD cassettes	*Take Me Beyond Your Leader*	McTernan
11:128	Code-tracking device	*Cross Reference*	Camp
11:172	Dump memory with new T-BUG	*T-BUG and Then Some*	Paxton
11:177	Eliminate volume proolems	*Up and Down*	Parris
11:206	Recapture a lost program	*Resurrect It*	Quindry
11:208	Add three loading instructions	*DOS Machine Code Loading Techniques*	Turner
11:226	PEEKing a directory	*You Can Call It Ray*	Kornfeld
12:147	Understanding Level II ROM	*Mysteries of the Level II ROM*	Griswold
12:160	Labeling and indexing routines	*Now It's Time for. . . Name That Tune*	Cornell
12:196	Robotics	*COMPAC*	Romanchik
12:212	EDTASM modification	*Assemble It Yourself*	Koch
12:257	Controlling data pointers	*RESTORE Data Pointer Control*	Cecil
12:259	Printing the display	*Less Is More*	Winterbauer
12:263	ROM vocabulary	*Keyword List Plus*	Decker

ADVERTISERS

RS Number		Page
81	AB Computers	74
282	ACR Consultants	57
452	A.M. Electronics	151
445	AT-80	114
229	Access Unlimited	125
34	Acorn Software Products	173, 195
97	Adventure International	135, 179, 154
387	Aerocomp, Inc.	119
88	Alpha Byte Storage	165
401	Alpha Products Company	29
262	Alpha Products Company	29
210	Alpha Products Company	29
138	The Alternate Source	189, 202, 249, 122
326	The Alternate Source	64
396	American Business Computers	98
387	American Business Computers	202
483	American Business Computers	225
484	American Business Computers	244
461	Ancle Labs	203
264	Apperat, Inc.	68
47	Applied Economic Analysis	156
*	Archbold Electronics	146
414	John Armstrong	259
148	Audio Video Systems	93, 138, 253
46	Automated Simulations	34
201	Barstrann Corporation	253
49	Basics and Beyond Inc.	156
184	Balden Corp.	63
351	The Berg Works	229
357	Big Five Software Company	175
377	Big Systems Software	225
444	Bitznbytes	114
235	The Bottom Line	120
6	The Bottom Shelf, Inc.	59
57	Bourrut Consulting Corp.	158
166	Harry H. Briley	58
393	CMS, Inc.	224
296	CPU Shop	91
145	C&S Electronics Mart Ltd.	171
294	Caldata Systems	235
62	Cecdat, Inc.	71, 219
46	Checks To-Go	111
459	Chicatrug News	179
32	Cload Megazine	101
100	CompuCover	117
107	Computer Applications Unlimited	254
199	Computer Case Company	221
372	Computer Discounts of America	201
22	Computer Information Exchange	231
178	Computer Program Associates	60
390	Computer Textile	209
321	Computers Unlimited	231
61	Computermat	254
392	Computex	244
415	Computex	237
9	Computronics, Inc.	138-145
204	Comsoft	222
10	Contract Services Associates	89
465	The Cornsoft Group	155
233	Cottage Software	229
160	Creative Computing Press	60
447	Creative Developments	118
119	Crown Plastics	124
*	Cryptext Corporation	158
7	Custom Computer Center	85
121	Custom Electronics	243
*	Cybernetics, Inc.	159
169	Cybernetics, Inc.	63
439	D-Soft	93
490	DFR Associates	243
371	Daltex	166
175	Data Access Corp.	64
44	Data Train, Inc.	97
274	Data Trans	148
453	Data Truss	136
436	Data Wholessle	258
302	Decision Master/Interlude	37
440	Discount Software Group	104
412	Discovery Games	259
68	Documan Software	72
488	E. F. Dreyer	243
253	Dynatek Information Systems Inc.	254
477	Edu-ware	243
*	Eighty Microcomputing	34, 50, 61, 161, 164, 262, 264, 265, 271
63	Elcompco	219, 255
339	Elcompco Microcomputer Peripherals	64
58	Electronic Specialists	130
26	Electronic Systems	267
278	Emtrol Systems Inc.	154
404	Epson America	79
40	Esmark, Inc.	157
3	Exstron	Cov. IV
141	FEC Ltd.	122
12	FMG Corporation	181
320	Fisher Associates	118
102	Fuller Software	202
254	Galactic Software Ltd.	209
475	GAMECRAFT	96
251	General Computer Company	65
79	Allen Gelder Software	235, 97
75	Godbout Electronics	80
218	Good Lyddon Data Systems	243
270	Mark Gordon Computers	161, 193
23	Hobby World Electronics	266
163	Howard W. Sams and Co., Inc.	60
103	Howe Software	254
37	IJO Inc.	206, 207
334	Information Technology Systems	84
300	Information Technology Systems	229
158	Image Computer Products Inc.	56
306	Insiders Software Consultants Inc.	211
2	Instant Software	38-41, 196-197, 199
329	Instant Software	58, 60
492	Integer Soft	246
379	Integrated Service Systems Inc.	253
246	Interface, Inc.	88
287	Interlude	17
295	Interpretive Education	72
187	International Software Assoc.	215
35	J. F. Consulting	254
315	JLS	189
249	JMS Corp.	126
160	JPC Products	177
155	JR Software	136
163	Joe Computer	88
65	Johnson Associates	231
149	Kogyosha Company	202
375	Krell Software	225
53	LNW Research	215, 148
450	LTM Inc.	272, 273
174	The Lawtech Co.	60
*	Level IV Products Inc.	163, 109
*	Liteboat Associates	239
471	Linnex Research Associates Ltd.	253
15	Lobo Drives International	Cov III
451	MTS Enterprises	72
87	Management Systems Software	211
60	Manhattan Software, Inc	130
156	Marigold Associates	72
165	McClintock Corp.	63
164	Measurement Systems and Controls	58
128	Med Systems Software	146
421	Medfield Computer Software	244
*	Mediamix	246
104	Mercer Systems Inc.	74
20	Meta Technologies Corp.	6, 7, 9, 11
54	Micro Architect	190
181	Micro Architect	63
214	The Micro Clinic	202
264	Micro Club	69
379	Micro-Design	256
182	Micro Developments Systems	60
476	Micro-80	249
89	Micro Learningware	263
72	Micro Management Systems Inc.	105
68	Micro Matrix	131
29	Micro Mega	149, 215
310	Micro Mint	221
485	Micro Mnemonics	254
177	Microperipheral Corp.	63
384	Micro Systems Software Inc.	153, 237
172	Micro Systems Software Inc.	58
341	Micro Systems Software Inc.	64
486	Micro Tax	256
95	Microcomp Software Systems	97
458	MicroCompatible Inc.	219
470	Microcomputer Systems	120, 235
28	Microcomputer Technology Inc.	68, 137, 185
307	Microcosm, Inc.	256
161	Microed	60
442	MICROGRAM	93
493	MICROGRAM	117
*	Micron, Inc.	80
360	Microtek, Inc.	21
8	Midwest Computer Peripherals	163
112	Miller Microcomputer Services	127
221	MISDSYS	223
*	Mullen Computer Products	239
144	Mumford Micro Systems	108
*	NRI Schools	167
142	National Tricor Inc.	224
164	New England Business Service, Inc.	167
116	Newby Software Development Company	117
243	Newtech Computer Systems Inc.	57
183	Northeast Microware	63
74	Northeast Microware	225
472	Northwest Micro Solutions	84
4	OK Machine & Tool	23
245	Okidata Corp.	27
389	Omega Sales	121
298	Orange Micro	192
86	PCD Systems	98
370	Pacific Exchanges	253
153	Pacific Office Systems	245
229	Palomar Software	256
64	Pan American Electronics	99
207	Pensadyne Computer Services	47
408	Percom Data Company	168
409	Percom Data Company	169
410	Percom Data Company	169
1	Percom Data Company	Cov. II, 169
43	The Peripheral People	70
51	Perry Gas & Oil	247
24	Personal Computer Systems	269
422	Personal Microcomputers Inc.	62
167	Persteve Electronics	60
170	Pharmacy Associates	60
17	The Program Store/Realsoft	106, 107
21	Programma International	81, 116, 200
364	The Programmer's Guild	179
441	Prosoft	243
395	QC Microsystems	171
269	Quent Systems	256
304	Quarp Publishing	124
41	Racet Computes	133
185	Radio Shack	58
*	Reality Software Company	253
433	Red Arrow Electronics	253
70	Remsoft Inc.	76, 189
276	Richcraft Engineering Ltd.	263
161	Rite 60 Software	64
468	Rochester Data Inc.	201
244	SJW, Inc.	209
*	S&M Systems Inc.	186, 190
261	Scientific Engineering Lab.	243
297	Service Technologies, Inc.	211
255	Michael Shrayer Software Inc.	75
19	Simutek	191, 205, 239, 154
87	Sirius Systems	86
30	Small System Software	192
232	Snapp, Inc.	13, 60
434	Soft Sector Marketing Inc.	115
173	Soft Sector Marketing Inc.	58
399	Software Efficiency	256
42	Software Etc.	123, 217
478	Software Innovations	64
286	The Software Mart	55, 30, 31
373	Southern Innovative Design (SID)	26
275	Speedway Electronics	246
154	Standard & Poors	19
455	Starrs-80	253
449	Sterling Computer Products	253
438	The Stocking Source	45
82	Sturdivant & Dunn, Inc.	96
150	Sublogic	189
151	Sun Research	229
266	Synapse Video	235
*	Synergistic Solar Inc.	202
358	Syracuse R & D Center	128
467	Systems-80	256
178	T.Y.C. Software	63
146	Tab Sales Company	224
45	Taranto & Associates	87
489	Tar Heel Software Systems Inc.	254
147	Task Computer Applicstons	263
25	Texes Computer Systems	235
*	Three-G Company Inc.	237
437	Tulsa Microsystems, Inc.	233
31	V R Data Corporation	77
432	Vern Street Products/Keyline Computer Products	201
374	Williams Enterprises	76, 249
355	Zocchi Distributors	111

***This advertiser prefers to be contacted directly.**

When It Comes To TRS-80 Add-on Memory...

LOBO Has It All.

LOBO DRIVES manufactures disk drive subsystems designed to provide TRS-80* users with a wide selection of low-cost, high-speed, efficient, mass-storge capabilities. Every LOBO DRIVES Memory System is thoroughly tested and burned-in to assure reliability and carries LOBO's unique one year, 100% parts/labor warranty.

Expansion and enhanced capabilities are key words in achieving full utilization of your computer system. LOBO DRIVES complete line of TRS-80 compatible disk drive subsystems is the ideal, cost effective way to provide the expansion capabilities you need to meet your system growth requirements

*TRS-80 is a trademark of Radio Shack A Tandy Company

TRS-80 MODEL II

LOBO DRIVES makes expanding your TRS-80 Model II very, very easy. Now you can add more floppy disk memory at less cost. And, LOBO can provide you with up to 40 MBytes of fixed disk Winchester technology storage capacity that is completely software compatible to your Model II.

- Model 800-850 8-inch dual Floppy Systems
- Model 1850 Dual Floppy/Fixed Disk Memory System

MODEL 1850 DUAL FIXED/FLOPPY DISK MEMORY SYSTEM

LOBO DRIVES has combined a 5 or 10 MByte Winchester technology fixed disk and 1.6 MByte double-sided, double-density floppy disk drive in one cabinet. The unique controller can accommodate two dual units. Now you can have the speed and reliability of fixed disk, with built-in floppy back-up.

- 5 or 10 MByte Fixed Disk Capacity
- Up to 1.6 MByte Floppy Disk Capacity
- Winchester Reliability
- Software Compatible

MODEL 800/850 DUAL FLOPPY DISK MEMORY SYSTEM

Complete with stylized cabinet, power supply, controller, interface, and cables, the Model 800/850 Dual Floppy Disk Memory System is the ideal way for the serious user to expand his disk-based TRS-80.

- Up to 3.2 MBytes Capacity
- Single-side, Single or Double Density
- Double-Side, Single or Double Density
- Complete Software Compatibility
- High Speed Access Time

MODEL 400 5¼-INCH FLOPPY DISK MEMORY SYSTEM

A low-cost, high performance, software-compatible Floppy Disk for TRS-80 Model I users

- Up to 220 KBytes Capacity
- Single/Double Density
- Soft Sector Format
- 298 Msec Access Time

MODEL LX80 EXPANSION INTERFACE

LOBO DRIVE's new Model LX80 expansion interface enhances system performance by expanding disk storage capacities beyond 40 MBytes, adding a second serial port and facilities for an additional 32 K RAM. The LX80 permits you to achieve the maximum expansion capabilities of your TRS-80

- Connects Directly to Keyboard
- Two Serial Ports (optional)
- One Parallel Expansion Port (standard)
- One Parallel "Centronics" Printer Port (Standard)
- Supports Double Density 5¼ and 8 inch Floppies
- Separate Port for 8-inch Floppies
- Switch for Overriding Keyboard ROM
- Separate Port for Fixed Disk Drives

MODEL 950 DUAL FLOPPY/FIXED DISK MEMORY SYSTEM

LOBO combines the outstanding capabilities of the latest technological breakthrough in disk drives, the Shugart Technology 5¼-inch Micro Winchester fixed disk drive with the proven reliability of the Model 400/450 Floppy Disk in one easy-to-use cabinet.

- The Storage Capacity of 16 double-sided, double-density Mini-Floppies
- Built-in Floppy Disk Back-up
- 170 Msec Average Access Time
- Sealed Environment/Winchester Reliability

NOTE: Limited Availability in the Fall, 1981

See your nearest dealer, call, or write for the complete LOBO DRIVES story find out just how competitively priced a quality drive can be

Pump Up Your TRS-80 with the ES/F Mass Storage System

▲ Actual Size Actual Thickness ▼

THESE FACTS SPEAK FOR THEMSELVES!

	CASSETTE	ES/F	MINI-DISK
SPEED (Seconds to load "Blackjack")	56	6 (5" wafer)	6½
CAPACITY (thousands of bytes)	38 (C-20)	64 (75" wafer)	59 (TRSDOS)
RELIABILITY (Designed for digital data?)	NO	YES	YES
SYSTEM COST (First unit plus interface)	$60	$250	$800
MEDIA COST (in quantities of ten)	$3.10 cassette	$3.00 wafer	$3.20 disk

Let's face it. Cassette players were not designed to store digital data and programs. That's why we designed a digital storage system using a continuous tape loop: the Exatron Stringy/Floppy (ES/F) and the Wafer. There's no expensive interface to buy—the ES/F comes ready to pump up your TRS-80.*

Once your TRS-80* is pumped up by our ES/F . . . you won't want to deflate it. We're so sure, that we offer an unconditional 30-day money-back guarantee and a one-year limited warranty. Over 2,000 TRS-80* owners have met the wafer . . . why don't you?

TRS-80*

EXATRON'S STRINGY/FLOPPY...

SPEED, CAPACITY AND RELIABILITY FOR ONLY $249.50

CALL
OUR HOTLINE
(800)-538-8559

IN CALIFORNIA,
CALL (408)-737-7111

exatron, inc.
[illegible] Commercial Street
Sunnyvale, Calif. 94086

*TRS-80 is a registered trademark of Tandy Corp.